Improving Histopathology and Medical Image Analysis with Deep Learning

EE 594 : Dual Degree Project Stage I and II

submitted in partial fulfillment of the requirements
for the degree of

Bachelor and Master of Technology
in Electrical Engineering

by

Jay Sawant
Roll No: 18D070050

under the guidance of

Prof. Amit Sethi

Department of Electrical Engineering
Indian Institute of Technology, Bombay
Mumbai - 400076.

2023

Dual Degree Project Approval

The dissertation entitled

Improving Histopathology and Medical Image Analysis with Deep Learning

by

Jay Sawant

(Roll No. : 18D070050)

is approved for the degree of

Bachelor and Master of Technology in Electrical Engineering

Digital Signature
Sharat Chandran (051054)
04-Jul-23 10:23:08 AM

Prof Sharat Chandran
Dept. of Computer Science and Engineering
(Examiner)

Digital Signature
V Rajbabu (i07164)
30-Jun-23 01:01:50 PM

Prof. Rajbabu Velmurugan
Dept. of Electrical Engineering
(Examiner)

Digital Signature
V Rajbabu (i07164)
30-Jun-23 01:01:58 PM

Prof. Rajbabu Velmurugan
Dept. of Electrical Engineering
(Chairperson)

Digital Signature
Amit Sethi (i17185)
29-Jun-23 06:34:04 PM

Prof. Amit Sethi
Dept. of Electrical Engineering
(Supervisor)

Date: June 21, 2023
Place: IIT Bombay

Declaration

I declare that this written submission represents my ideas in my own words and where other's ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will cause disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Date: 21 June, 2023

Jay Sawant

Roll No. 18D070050

Digital Signature
Amit Sethi (i17185)
29-Jun-23 06:29:09 PM

Abstract

The advent of deep learning models and algorithms in the field of Histopathology, coupled with the utilization of Whole Slide Images (WSIs), necessitates the implementation of a Quality Control (QC) mechanism to ensure the accuracy and reliability of the models' performance. In Chapter 1, we present HistoROI, a ResNet18-based classification model designed to classify patches within WSIs into six distinct pathology-relevant regions of interest (ROIs): epithelial, stroma, lymphocytes, artifacts, miscellaneous, and adipose. HistoROI is trained using a human-in-the-loop active learning paradigm, which incorporates diverse training data to enhance generalization capabilities. To evaluate the efficacy of HistoROI, we compared its performance with a widely used QC tool known as HistoQC, specifically focusing on artifact detection tasks. Through experiments conducted on a dataset comprising 93 annotated WSIs, HistoROI demonstrated superior performance in comparison to HistoQC. Additionally, we enhanced the training procedure of the HistoROI classification model by incorporating the Supervised Contrastive Learning technique.

In Chapter 2, I present the findings of my internship project, where I explored the application of the Supervised Contrastive Learning method for opacity detection in Chest X-rays. The task involved binary classification to determine the presence or absence of opacities in the images. My experiments revealed that employing Supervised Contrastive Learning instead of conventional training methods significantly boosted the model's performance, demonstrating improved accuracy and efficiency in opacity detection.

We present the findings of Chapter 3, focusing on the Cell detection problem within the Ocelot 2023 challenge. Traditionally, cell detection methods solely rely on training models using zoomed-in patches containing cells. However, the Ocelot challenge dataset provides us with additional information, including zoomed-out tissue patches along with annotations. Through our observations, we demonstrate that our model, which considers a broader tissue-level context in conjunction with the input patch for cell detection and classification, outperforms conventional models trained solely on cell patches. This highlights the importance of exploiting cell-tissue interactions when addressing cell detection and classification, emphasizing the need to consider the broader context for improved results.

In Chapter 4, we address the critical issue of quality control in Whole Slide Images (WSIs) using advanced deep learning techniques. Our study focuses on developing multiple segmentation models based on the U-Net architecture to accurately detect and segment various artifacts, including pen markings, blur levels, tissue folds, tissue regions, and fat. To train these models, we leverage the HistoROI framework to generate a dataset that eliminates existing biases in WSIs. The performance of our models is evaluated by comparing the resulting usable masks with the widely used quality control tool, HistoQC, yielding a high level of agreement between the two approaches. Additionally, we assess the blur level segmentation model on two publicly available datasets, TCGA@Focus and Focuspath. These findings highlight the effectiveness of our deep learning-based approach in addressing quality control challenges in WSIs, providing valuable insights for improving the accuracy and reliability of histopathological image analysis.

Acknowledgments

I express my sincere gratitude towards my guide **Prof. Amit Sethi** for his constant help, encouragement and inspiration throughout the project work.

I would like to thank the research scholar **Abhijeet Patil** and M.Tech student **Harsh Diwakar** for constant support and guidance towards my contribution to this project and for providing the necessary resources required to complete my Stage-1 and Stage-2 projects.

Last but not least, I would like to thank the whole **MeDAL** family for always being helpful in times of need.

Jay Sawant
IIT Bombay
June 21, 2023

Contents

List of Figures

Chapter 1

Deep learning based automatic patch level segregation model and its use in Quality Control of WSIs

1.1 Introduction

With the advancements in digital pathology came the technique of digitizing a tissue slide in an image format called as the Whole Slide Imaging (WSI). This technique makes it easier for pathologists to scroll and zoom in on an area of interest in the image. Since these WSI images have a resolution that may go upto $0.25\mu m$ per pixel i.e. the height and width of the image may go upto 1 million pixels, it becomes difficult to analyze the enormous images. Hence, these images are usually analyzed by looking at their small patches at multiple zoom levels. While analyzing the patches, it is found that a significant number of patches are not usable due to many reasons like the patch being from background or the patch consisting of some artifacts making it non-usable and other reasons. This arises the need of keeping a Quality Control check on the Whole Slide Images. Many deep learning models are trained using WSIs in various problem statements. Hence, it becomes necessary to have a pre-processing tool so that we can feed only the useful and relevant regions in the Whole Slide Images to the models to have a better performance.

1.1.1 Potential artifacts in a Whole Slide Image

Artifacts in a WSI can be intrinsically present within the tissue, may get introduced in the slide preparation process and its digitization or as a result of ageing or long term slide storage. Examples of artifacts introduced during the slide preparation process include variability in tissue section thickness, tissue folding, variability in H&E staining, air bubbles and/or dirt under the coverslip, pen-markings, old glass. Artifacts introduced while digitization of the slide occur mainly due to the scanner being out of focus generating blurry images or the jpeg compression algorithm. The most common histo-

logical artifacts can be seen in the Fig 1.1. It is evident from the literature review below that the presence of artifacts in the WSI dataset downgrades the performance of many models. Hence, it becomes essential to have a Quality Control check as a pre-processing unit in the routine usage of histological images.

1.2 Review of Literature

1.2.1 Quality Control Stress Test

The paper[1] explores the influence of artifacts in Whole Silde Images on the performance of the pre-trained, validated deep-learning based model for prostate cancer detection. They produce the most common artifacts synthetically by digital means and perform a systematic stress test for Deep-Learning based model for prostate cancer (PCA) detection.

Datasets

For the stress test purpose, the authors of [1] used six different datasets from four institutions, all digitized using different scanner systems. To reduce the computation cost, a random crop of 120,000 patches (all patches are classified by the model correctly) from each dataset was generated, each crop consisting of 50,000 patches with tumor tissue, 50,000 patches with nonneoplastic glandular prostate tissue, and 20,000 patches with nonglandular tissue.

Model Description

The Deep Learning-based patch-level classification model for prostate cancer detection proposed in [6] was used in this study. The model takes a 300x300 patch as an input and has a InceptionResnetV2[7] architecture with a classification head to classify three classes named prostate glandular tissue, nonglandular tissue, and tumor tissue.

Artifacts Generation

The most common artifacts in the routine histopathology practice were synthetically generated. These include focus, elastic deformation, brightness, contrast, dark spots (e.g., dust, cover glass scratches, and other kinds of contamination), synthetic threads overlying tissue, contaminating squamous epithelial, greasy fingerprints on the slide surface, and H&E staining scheme (Fig 1.1). Other artifacts related to digital processing of the image like jpeg compression, rotation of patch, and flipping of patch were also synthesized.

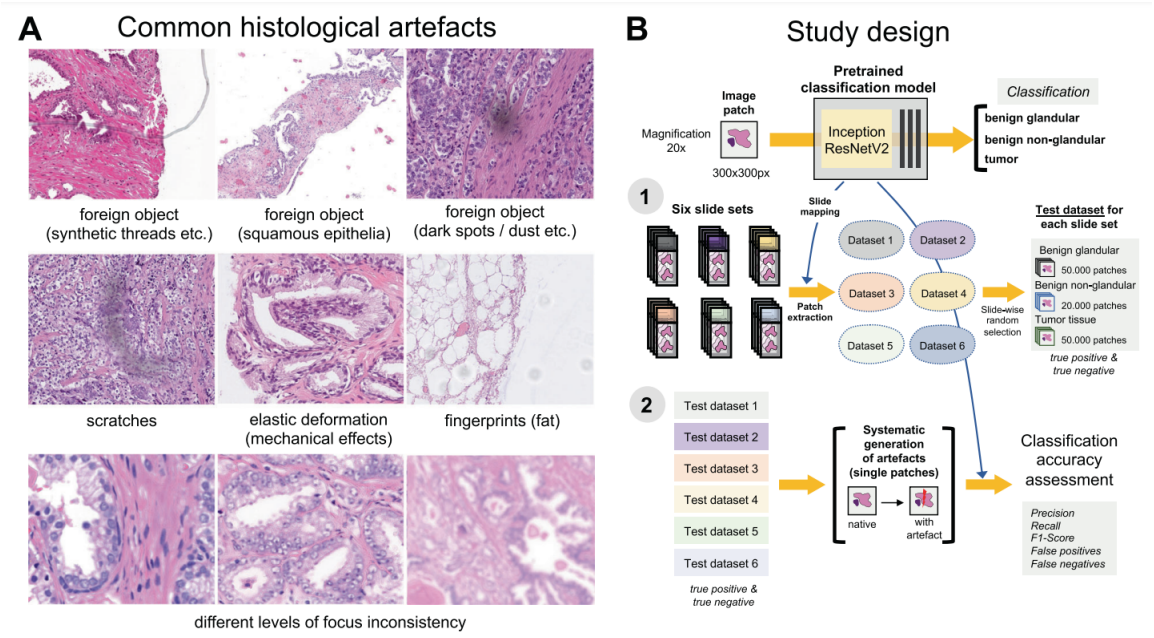


Figure 1.1: Common histological artifacts and stress test design [1]

Testing Pipeline and Results

During the stress test, each artifact was introduced in all the patches of all 6 datasets maintaining the variability in each artifact and model classification was carried out on these modified patches to estimate model accuracy in presence of artifacts. The results of the test of all datasets were analyzed and summarized with regard to false positive (benign tissue classified as tumor) and false negative results (tumor tissue classified as benign) showing the impact of each artifact on the model performance which correctly classified all the patches before inducing artifacts (Fig 1.2)

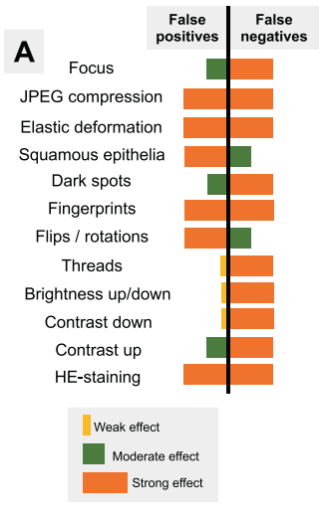


Figure 1.2: Analysis of artifact-induced misclassifications [1]

1.2.2 The Effect of Quality Control on Accuracy of Digital Pathology Image Analysis

The paper [2] discusses various Quality Control issues which are introduced right from the start of the slide preparation upto the image digitization stage. This work aims to determine the extent to which image quality issues affect the automated analysis. This study attempts to assess the image quality of a digital slide dataset from a clinical trial and provides a comprehensive view of how image quality affects subsequent analyses.Their experiment design flowchart is represented in Fig 1.3. The experiment is divided into 6 sections as follows:

1. Colon cancer data is collected through clinical trial over 5 years. For each of the 2211 colon cancer cases scanned, tumours were identified by a train pathologist and the tumour region was annotated. Random 50 points were generated from the annotated region and each point was labelled either tumour or stroma.After removing non-informative data points, the dataset totaled 106268 pathologist-scored x-y co-ordinates

2. The Machine Learning algorithm published in [8] was used for training and testing on the above dataset. It used random forests algorithm[9] to learn an optimized minimal set of features derived from a patch of 256x256 pixel area surrounding the center of each labelled x-y coordinate. Once the random forest was trained, predictions were made for each x-y co-ordinate and Tumour-Stroma Ratios (TSRs) for each slide were generated per-case.

3. The top 100 worst cases after testing were picked out for manual checking of Quality Control issues. Most of the observations consisted of artifacts created due to variability in staining levels. The authors predicted that applying image analysis to a dataset free from these artifacts would improve the model performance.

4. The pathologists were required to visually inspect all 2211 colon cancer cases and apply the single most appropriate classification to each case.

5. Based on algorithm-pathologist TSR differences, algorithm performance was assessed using the clinical trial dataset that had been labelled by a pathologist. To determine how and to what degree the algorithm is impacted by the dataset issues, observations were made using the QC categories with the biggest differences.

6. The algorithm was retrained and tested using both the full dataset and the AC accepted cases only. Using the same methodology as the original algorithm, 10 fold cross validation was performed and the accuracy was recorded for 4 combinations: training and testing on all the data; training and testing on the QC dataset only; training on the QC dataset and testing on the whole dataset; training on the whole dataset and testing on the QC dataset

The results (Fig 1.4) indicate that quality issues clearly affect performance of automated solutions to varying extents, and need to be compensated for either prior to processing, or as part of algorithm design, in order to avoid error in processing routine digital slides

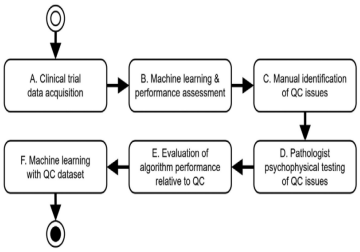


Figure 1.3: Experiment Design Flowchart [2]

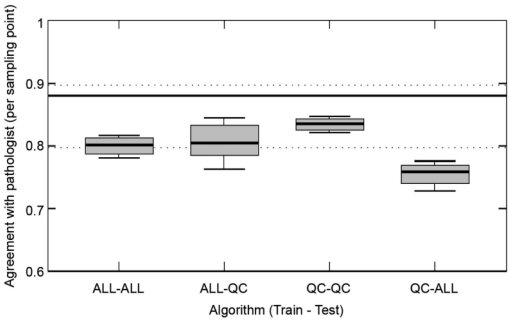


Figure 1.4: Effect of removing rejected QC images on algorithm accuracy [2]

1.2.3 HistoQC: An Open-Source Quality Control Tool

Introduction

Various types of artifacts get introduced in the final Whole Slide Image (WSI) due to small unavoidable errors made during the process of slide preparation and its digitization. Manual review of glass and digital slides is laborious and have a high variability subject to different pathologists. Hence, the authors of [10] developed a tool which ensures a reproducible automated approach of precisely localizing artifacts to identify slides that need to be reproduced or regions that should be avoided during computational analysis.

Methods

To run the HistoQC tool, the user supplies a configuration file consisting of parameters. The important ones include which modules should the tool run and in what order, from what level of the WSI should the image be extracted for analysis, different kernel sizes and thresholds for different modules and other parameters. Once all the modules mentioned in the configuration file are executed by the python-based pipeline, relevant output images are created which includes thumbnail of the WSI, a mask indicating the useful region in the WSI, a mask pointing at the blurry locations in the image, a mask showing if there are any pen markings in the image and so on.

Results and Discussion

Two pathologists with experience in digital pathology were asked to grade each of the output masks of HistoQC as either acceptable or not acceptable in order to validate the results provided by the HistoQC tool. A minimum 85% area overlap between the pathologists' visual evaluation and HistoQC's computational evaluation of artifact-free tissue was required to be considered acceptable. Overall, there was 95% (477 of 500) agreement between HistoQC and the experts.

1.2.4 PathProfiler: Automated Quality Assessment of Retrospective Histopathology WSI Cohorts by AI

The paper [3] proposes a a quality assessment pipeline in which possible multiple artifacts are predicted in a same region along with diagnostic usability of the image. A multi-task deep neural network is trained to predict if an image tile is usable for diagnosis/research and the kind of artifacts present in the image tile. Quality overlays are then generated from image tile predictions. Quality overlays are further mapped to a standard scoring system to predict the usability, focus and staining quality of the whole slide images.

Dataset Annotation

A subset of Prostate Cancer cohort was annotated by specialist urological pathologists to create a dataset with labels for training and testing purpose. This dataset consisted of 107 H&E stained WSIs of prostate tissue (biopsies and TURPs) from the ProMPT cohort and 91 H&E stained WSIs of contemporary prostate biopsy cases. The 198 slides provided a manually selected dataset of 1711 annotated image patches which was divided into the training set (80%), test set (10%), and validation set (10%) for patch level classification model, and the dataset of 198 annotated whole slides which was divided into the training set (60%) and test set(40%) for the models predicting the slide level quality score.

Multi-label Model Training

The training pipeline that the PathProfiler follows is depicted in Fig 1.5 A pre-trained Resnet-18 architecture [11] was used which takes 3-channel input images at a size of 224×224 pixels for the multi-label model training. The last fully connected layer was modified to output six classes with linear activation functions.
Dataset used for the training consisted of 1711 patches (split as 80%,10%,10%) with labels annotated by specialist urologists as per the Fig 1.6.
Since noisy labels in the dataset were inevitable, the authors of [3] employed the Huber loss Function[12] (with δ=1) which is robust to label noise. The Huber Loss function is as follows:

$$L_\delta(x, y) = \begin{cases} \frac{1}{2}(x-y)^2 & \text{for } |x-y| \leq \delta \\ \delta|x-y| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases} \tag{1.1}$$

The model was trained for 200 epochs with a batch size of 100, learning rate of 1e-4 and a weighted batch sampler to handle the label imbalance.

Once this model is trained, on passing all the patches from tissue region through this

model, a quality overlay is generated for each output category. In the next step, we map the predicted quality overlays to the slide-level standardised scoring system. For this, statistical parameters of quality overlays are used to predict slide-level quality scores; overall usability of the WSI (binary 0 or 1), and a score 0-10 for quality of focus and H&E staining from the lowest quality to highest quality, where the cut-off score for acceptable quality for diagnostic purposes is 4. The labels to train these models (that predict the slide level quality scores) are provided by the urologists for each slide (train-60%, test-40%)

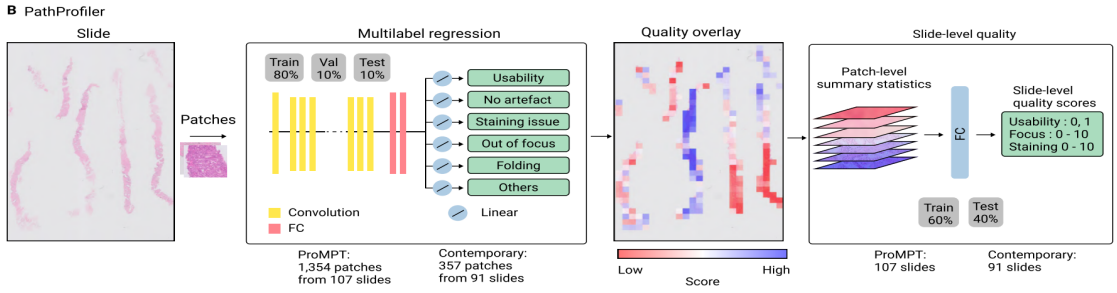


Figure 1.5: PathProfiler quality assessment pipeline [3]

Label	Criteria	Value
y_1	Usability	1 - appropriate for diagnosis 0 - otherwise
y_2	No artefact	1 - no presence of any slight or severe artefacts 0 - otherwise
y_3	Staining artefacts	1 - severe staining or H&E contrast issues 0.5 - slight staining or H&E contrast issues 0 - no staining or contrast issues
y_4	Focus artefacts	1 - severe focus artefacts 0.5 - slight focus artefacts 0 - no focus artefacts
y_5	Tissue folding	1 - the presence of tissue folding 0 - otherwise
y_6	Other artefacts	1 - the presence of other artefacts such as dirt, glue, ink, cover slip edge, diathermy, bubbles, calcification and tissue tearing. 0 - otherwise

Figure 1.6: Patch level labels [3]

Results

The model performance on the test dataset is shown in Fig 1.7 and Fig 1.8. The Table F in Fig 1.8 shows a very high correlation between the metrics defined by slide level predictions and the scores provided by the urologists for the overall usability, focus and staining of all the WSI images.

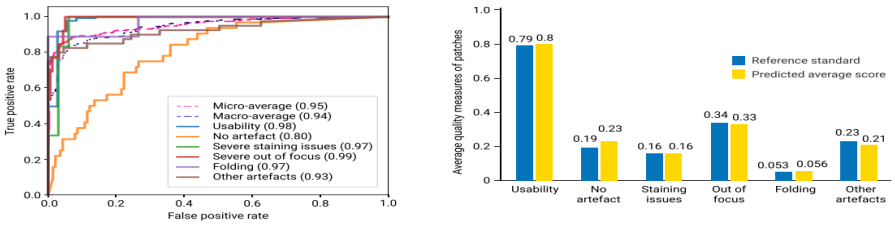
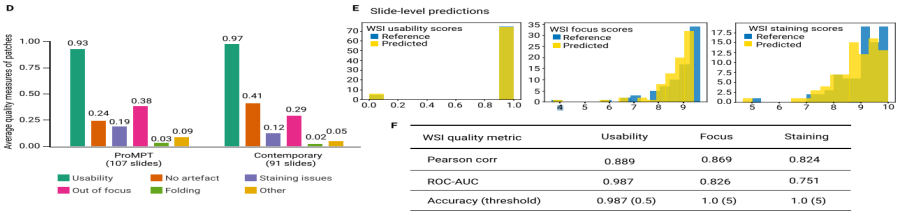


Figure 1.7: Model performance for test dataset of image patches [3]



F

WSI quality metric	Usability	Focus	Staining
Pearson corr	0.889	0.869	0.824
ROC-AUC	0.987	0.826	0.751
Accuracy (threshold)	0.987 (0.5)	1.0 (5)	1.0 (5)

Figure 1.8: Average Quality measures of patches and slide level predictions[3]

1.3 HistoROI: Histopathology-specific preprocessing

A Whole Slide Image is a digitized version of a tissue section carefully taken out from a particular organ. The tissue section consists of multiple type of cells like epithelial, stroma, lymphocytes and so on. Identification of these cells in the WSI sometimes become essential in various diagnostic tasks. In this section, we introduce the HistoROI - a ResNet18-based classifier to segregate WSI into six broad pathology types - epithelial, stroma, lymphocytes, artifacts, miscellaneous and adipose. A human-in-the-loop active learning approach is used to train the HistoROI, ensuring variations in the training data for better generalisation.

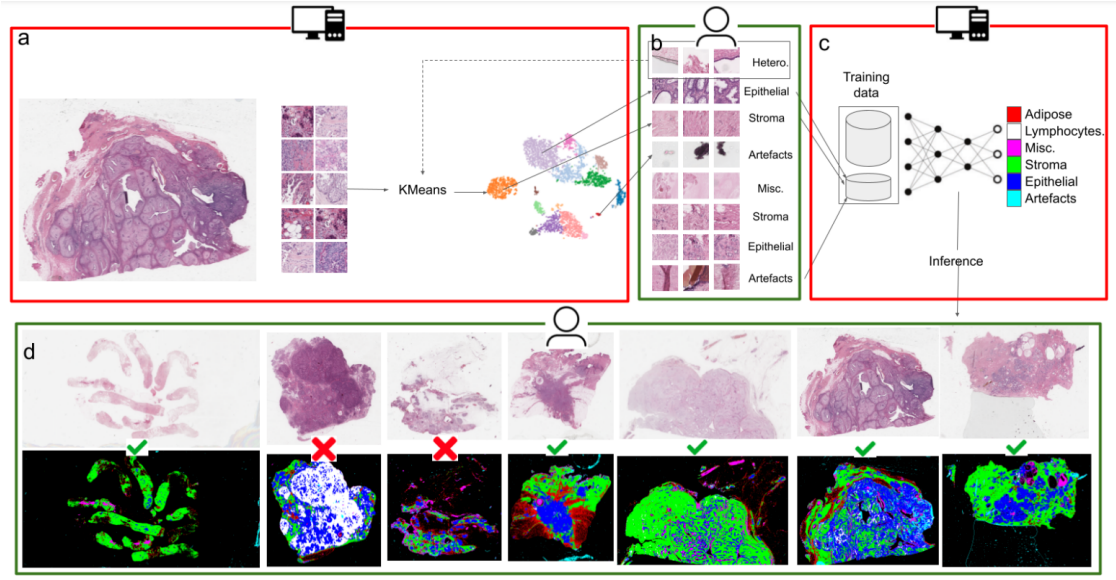


Figure 1.9: Human-in-the-loop training pipeline for HistoROI. Actions in red boxes are automatic, and actions in green boxes are manual. (a)Embeddings of the patches of WSI are divided into clusters. (b) Clusters are manually annotated. Heterogeneous clusters are re-clustered(shown in dotted line) (c) Annotated data is added to previously annotated data and HistoROI is trained with updated data. (d) The trained model is inferred on multiple WSIs. WSIs with poor performance are manually identified and annotated in the next iteration of training.

1.3.1 Datasets

For the training of the HistoROI, patches are carefully selected from 50 WSIs in the BRIGHT[13] dataset to capture variations for better generalisation. The dataset consists of more than 2 million patches from these 50 WSIs.
In order to validate the performance of HistoROI on correctly identifying the 'artifacts' class, a dataset of 93 WSIs from TCGA data portal is created by manually annotating the foreground tissue region in all WSIs. This dataset contains tissues slides from the organs of breast, lung, kidney and prostate with almost equal number of slides from each organ.

1.3.2 Methods

Annotation of a single WSI

Patches are extracted from the 10x magnification of a WSI with size 256x256. The patches with more than 95% of the pixels having average pixel value more than 230 are discarded. These patches are passed through the EfficientNet-B0 model[14] and 40-dimensional feature vectors are obtained for each patch from the 1^{st} block of the model using average pooling. The features obtained are then passed through the K-means clustering algorithm to make 32 clusters. By visualizing 25 random samples from each cluster, the cluster is assigned one of the class from 6 classes (1. Epithelial 2. Stroma 3. Miscellaneous 4. artifact 5. Fat and 6. Lymphocytes). If the samples from a particular cluster does not fall in any of the above 6 classes, it is called as a heterogeneous cluster. Features of patches from all the heterogeneous clusters are again passed through the K-means algorithm to make 32 clusters and each cluster is assigned a label). The clusters which remain heterogeneous at this point are now discarded.
An initial dataset of 20 WSIs is created by annotating each WSI according to the above procedure.

Human-in-the-loop training

In our case, the dataset contains many data points with similar features. Hence, by manually annotating each WSI, we are not utilizing the efforts of data annotators optimally. Hence, to address this problem, we train a model which helps in the annotation of new WSIs with Human-in-the-loop approach.
We initially train a Resnet-18 based six-class classifier with the data from 20 WSIs (15 WSIs for tranining and 5 WSIs for validation). Cross Entropy loss is minimized with Adam optimizer[15] and the model with the least validation loss is inferred on all the WSIs from the BRIGHT dataset. WSIs along with their predictions are visually analysed using QuPath and the ones with poor performance indicate that these WSIs are out of the distribution of the training dataset. Hence, these WSIs are then annotated and added to the training dataset for further fine-tuning of the classifier.

This cycle is repeated for 3 times adding 10 new WSIs each time into the training set. Finally, we have a training dataset from 50 carefully selected WSIs which contain enough variation for generalisation. The whole process of Human-in-the-loop training is summarized in Fig 1.9. Final HistoROI six-class classification model is then trained using these 50 WSIs.

1.3.3 HistoROI as a Quality Control Tool

To identify the performance of HistoROI on artifact prediction, 93 WSIs from the TCGA data portal[16] of four different organs (breast-27, lung-21, kidney-21, prostate-24) were hand-annotated by our pathologist. We created a patch-usability mask for each WSI using the predictions of the HistoROI model (patches classified as artifacts and adipose are not usable) and compared it with the usability masks obtained from running the popular HistoQC[10] tool.

Problems with HistoQC tool

The default magnification level of the WSI image used by the HistoQC tool for running its model is (1.25x). Hence, the HistoQC tool extracts the working image from higher levels of the WSI file using Openslide library considering that the level 0 corresponds to the image of highest magnification. But, when images are extracted from the higher levels of the WSI, most of these images contains checkerboard like severe artifacts as shown in Fig 1.10. Hence, the output results generated by the tool become meaningless. We modified the source code of HistoQC such that the working image is extracted from the level 0 (purest form of the image) and then resized it to the target magnification level yielding a clearer image.

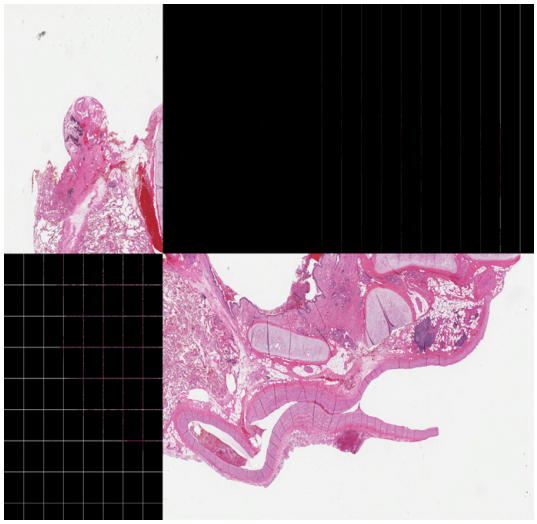


Figure 1.10: WSI thumbnail used by HistoQC tool

Figure 1.11: Use-Mask generated by the HistoQC tool

Comparison of HistoROI with HistoQC

We inferred the 93 WSIs using our HistoROI tool and created binary masks for each WSI. The $0's$ in the mask correspond to the prediction of HistoROI made as 'artifact' or 'adipose'. All the pixels in a patch were labelled as 0 or 1 as per the prediction to create the binary mask. The masks were then resized to the size corresponding to the masks obtained by hand-annotation. Same was performed for the masks obtained from HistoQC tool. The mean Dice score over WSIs between HistoROI and hand annotations is observed to be 0.87 whereas, for HistoQC, it is observed to be 0.83. A few qualitative results are shown in Fig 1.12. The performance of HistoROI is better on 65 WSIs out of 93 total WSIs. Comparison in the form of scatter plot is shown in Fig 1.13

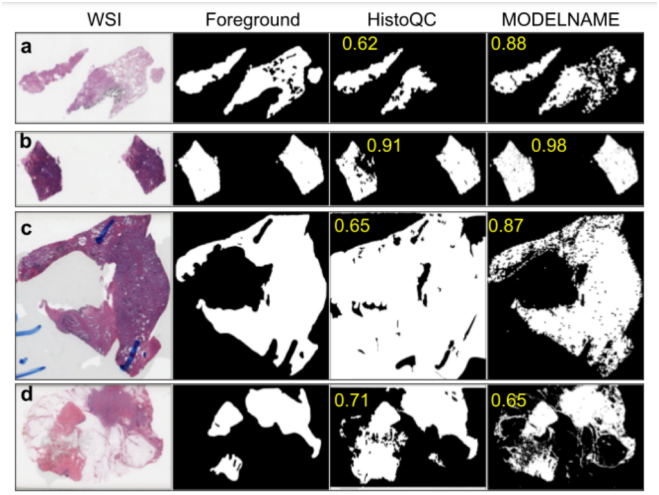


Figure 1.12: A few samples of WSIs with foreground detected by HistoQC and HistoROI. All the values in the diagram are Dice scores

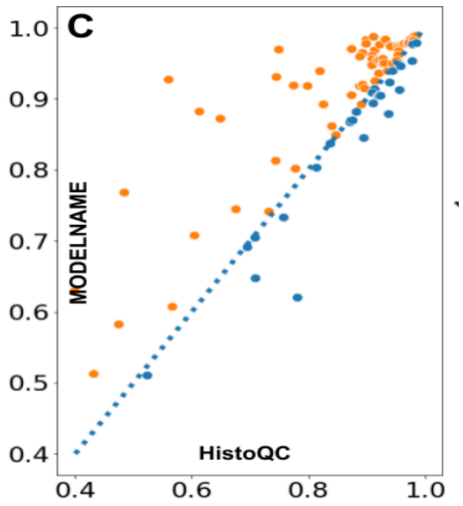


Figure 1.13: Scatter plot of Dice scores. The blue dotted line indicates $y = x$ line

Analysis of the above comparison

According to our observations, HistoQC tends to identify the region with relatively less dense tissue as fat (Figure 1.12-a). Also, HistoQC fails to distinguish between foreground and background when background pixels are greyish. Because of this, HistoQC performance degrades in the presence of air bubbles (Figure 1.12-b), coverslip-related artifacts (Figure 1.12-d), etc. HistoROI performs better compared to HistoQC for the above-mentioned scenarios. On the other hand, HistoQC detects pen marks better than the proposed model. Further analysis of training data for HistoROI showed that it contains only one WSI with pen marks. These pen marks are also observed to be outside the tissue region. Hence, in Figure 1.12-c, pen marks outside the tissue region are correctly identified as background by HistoROI whereas performance is not is expected for pen marks which are on the tissue region. Though the predictions of HistoROI are pixelated, because of the patchbased model, the overall Dice score is better than HistoQC. This indicates that the patch-based dataset contains enough variation for artifacts.

1.3.4 Improvisation of the HistoROI classification model

The HistoROI six-class classification model was trained using the patches from the 50 WSIs as mentioned in section 1.3.2. The training data consisted of patches from all 50 WSIs and the model was trained using the conventional supervised training algorithm[17] with Cross-Entropy loss optimization. The output layer of this model is a fully-connected layer taking a input of 512 dimensional feature vector and outputs a 6 dimensional vector corresponding to the 6 classes. By using this conventional training method, the cross-entropy loss tries to find decision boundaries between these 6 classes without considering the location of the 512-dimensional feature vectors of same class. Ideally, it is expected to have the feature vectors of same class close to each other and feature vectors with opposite labels far away from each other in the feature space. But, optimizing the cross-entropy loss only does not help in achieving the same. To address this problem, we use the Supervised Contrastive Learning[18] to train our model.

Supervised Contrastive Learning

As an overview of this learning method, during training, a batch of n images is considered. For each image in the batch, two views of the same image are created using transformations like random crop, horizontal/vertical flip and so on. These views for all images are combined to form a batch of $2n$ views. Now, loss function of this method is such that feature vectors of all the views belonging to the same class will be brought closer to each other and feature vectors of views belonging to different class will be sent far apart. The Loss function used by this method is given by Eqn 1.2

$$\mathcal{L}_{\text{out}}^{\text{sup}} = \sum_{i \in I} \mathcal{L}_{\text{out},i}^{\text{sup}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau\right)}{\sum_{a \in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau\right)} \tag{1.2}$$

Here, $i \in I \equiv \{1 \ldots 2N\}$ be the index of an arbitrary augmented sample, $A(i) \equiv I \backslash \{i\}$, $P(i) \equiv \left\{p \in A(i) : \tilde{\boldsymbol{y}}_p = \tilde{\boldsymbol{y}}_i\right\}$ is the set of indices of all positives in the multiviewed batch distinct from i, and $|P(i)|$ is its cardinality.

Improved Loss function of the HistoROI classification model

Previous architecture of HistoROI consisted of Resnet18 as backbone that outputs a 512-dimensional feature vector and a single layer linear classifier at the end for 6 classes. The Loss function used is described in Eqn 1.3.

$$\mathcal{L}_{CE} = \sum_{j=1}^{N} L_j = -\sum_{i=1}^{6} y_i log(p_i) \tag{1.3}$$

Here, y_i is the ground truth, p_i is the softmax probability and N is the batch size.

In the current architecture, along with the Resnet18 as backbone, two two-layer heads are added. The first head is called the projection head which takes the 512-dimensional vector from the backbone and outputs a new 512-dimensional vector. The second head is the classifier which takes the feature vector from backbone as the input and outputs the probabilities corresponding to the 6 classes.

Now, the output from the projection head is used for minimizing the supervised contrastive loss whereas the output of the classification head is used to minimize the cross-entropy loss.

The final Loss function used for the training is described in Eqn 1.4

$$\text{Total Loss} = \mathcal{L}_{\text{out}}^{\text{sup}} + \lambda \mathcal{L}_{CE} \tag{1.4}$$

Here, λ is a hyper-parameter and chosen to be 0.5 in this case. For the training purpose, from the available 50 WSIs, 40 WSIs were chosen randomly to form a train set and rest of the 10 WSIs for validation. The model was trained for 50 epochs with a batch size of 128 with each batch having samples from all classes. Patches of size 256x256 from the 10x magnification level were used as inputs. To create the two views of each patch, transformations like random horizontal/vertical flip, ColorJitter, Random-Affine and Gaussian blurring were used. For optimization of loss, SGD optimizer was used with learning rate 0.005 initially and a lr-scheduler was used to reduce the learning rate by monitoring the validation loss.

1.4 Results and Discussion

T-SNE plots were used to observe the quality of learnt representations using the Supervised Contrastive Learning method. To compare the quality of the 512-dimensional representation vectors between the original HistoROI model and the model trained using the above loss, the quality of clusters in the T-SNE plots was compared as shown in the Fig 1.20. This analysis was done on the validation set of 10 WSIs with avaialble ground truth labels. The colour scheme map used in the plots to represent classes is as follows: Green-Epithelial, Red-Stroma, Blue-Scattered Stroma(adipose), Purple-Lymphocytes, Pink-Miscellaneous, Golden Yellow-artifacts

Observations:

1. In the Fig 1.14, we see that the variance of the lymphocytes embeddings is high. Also, many lymphocytes embeddings are found in the clusters of epithelial and stroma. The new HistoROI model overcomes such problems by tight clustering of the embeddings and better decision boundaries as seen in Fig 1.15. Similar things can be observed in the other images 1.20, where the original models tends to have a loose clustering since the cross entropy loss focuses only on finding good decision boundaries as opposed to the new model which aims at finding good decision boundaries but also maintains the quality of the cluster because of the contrastive loss.

2. Since, the dataset preparation pipeline is such that patches in a cluster (section 1.3.2) are labelled based on visual information from 25 random samples from that cluster, the dataset is expected to have noisy labels. This is observed in the clusters formed by the representation vectors of the new HistoROI model. For example, in Fig 1.25, the patch is labelled as 'stroma' and found in stroma cluster but it contains sufficient epithelial cells. In Fig 1.26, the patch is labelled as 'epi', it contains 'epi cells too, but is found in stroma cluster. This means there is such noise in the training data also which results in such ambiguity. There are also

some patches who have ground truth label as stroma and scattered stroma but are found inside the epithelial cluster. On visualizing these patches (Fig 1.21, 1.22,1.23, 1.24), it was observed that the patches contained sufficient amount of epithelial cells and hence were pulled towards epithelial cluster. This asrises a problem described in the section 1.4.1

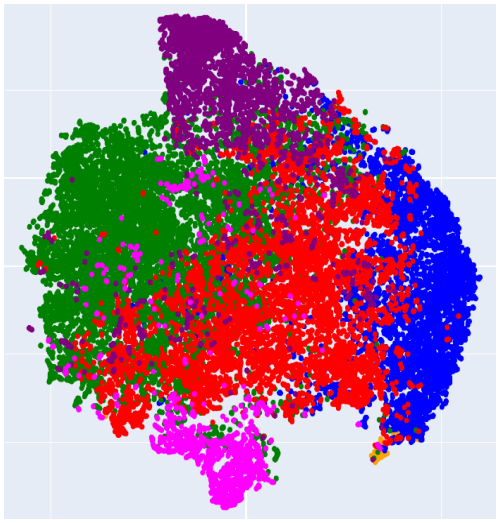


Figure 1.14: Representations of original HistoROI (BRACS_1492.svs)

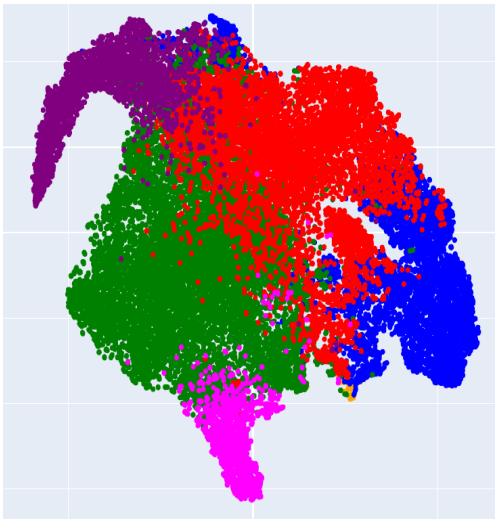


Figure 1.15: Representations of HistoROI with SupCon Loss (BRACS_1492.svs)

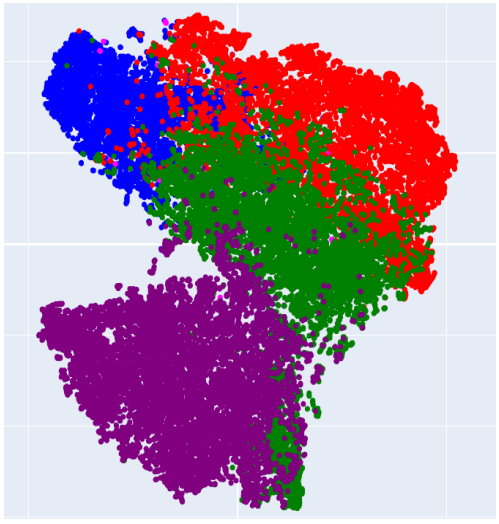


Figure 1.16: Representations of original HistoROI (BRACS_1637.svs)

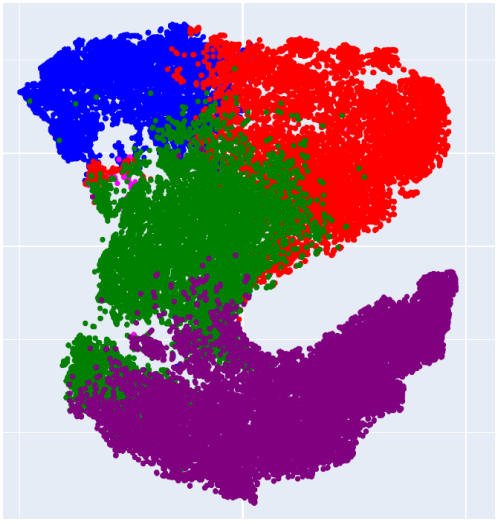


Figure 1.17: Representations of HistoROI with SupCon Loss (BRACS_1637.svs)

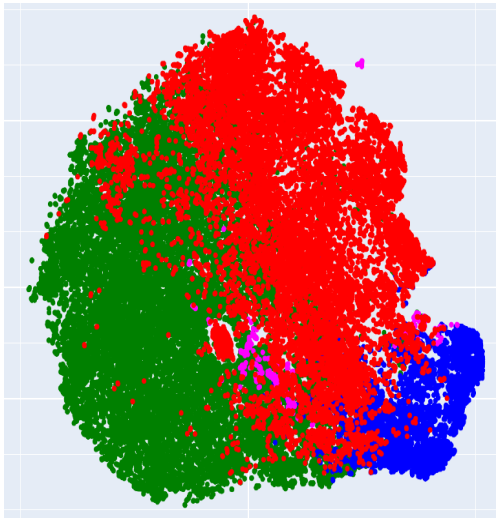


Figure 1.18: Representations of original HistoROI (BRACS_1918.svs)

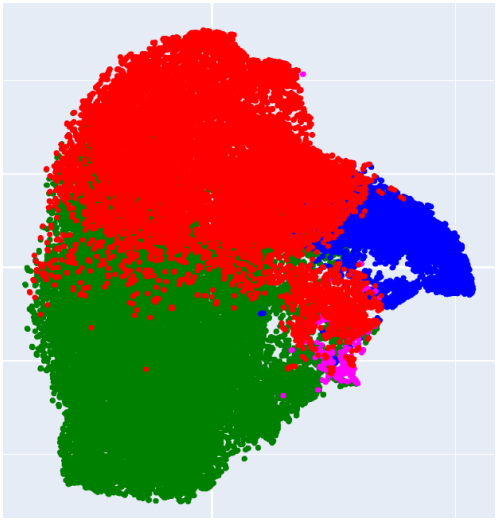


Figure 1.19: Representations of HistoROI with SupCon Loss (BRACS_1918.svs)

Figure 1.20: Comparison between the cluster quality of the representations of the original model and the representations of the model trained using SupCon Loss

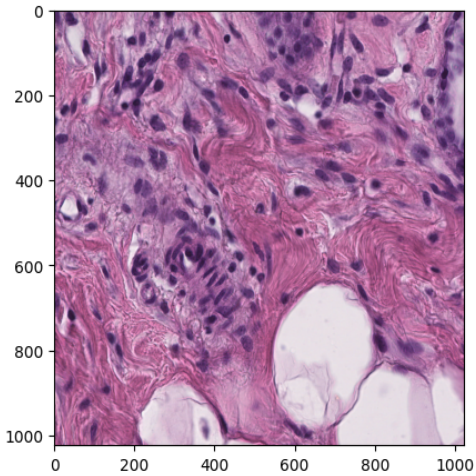


Figure 1.21: Labelled as stroma but found in 'epi' cluster

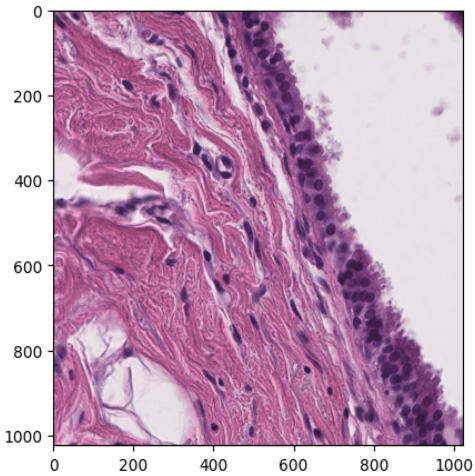


Figure 1.22: Labelled as stroma but found in 'epi' cluster

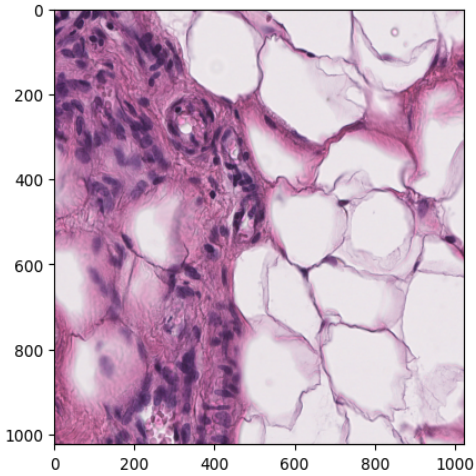


Figure 1.23: Labelled as scattered-stroma but found in 'epi' cluster

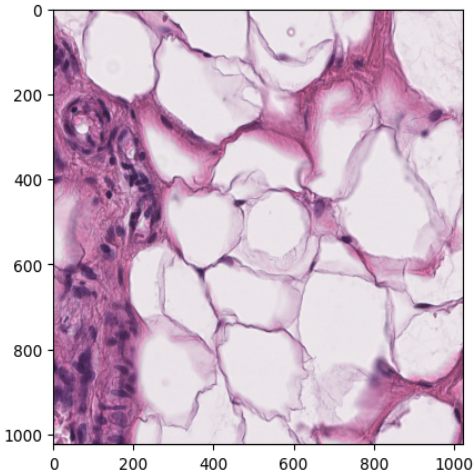


Figure 1.24: Labelled as scattered-stroma but found in 'epi' cluster

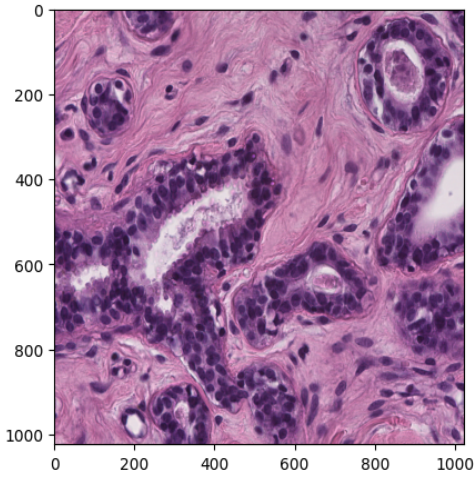


Figure 1.25: Patch in 'stroma' but contains 'epi' cells

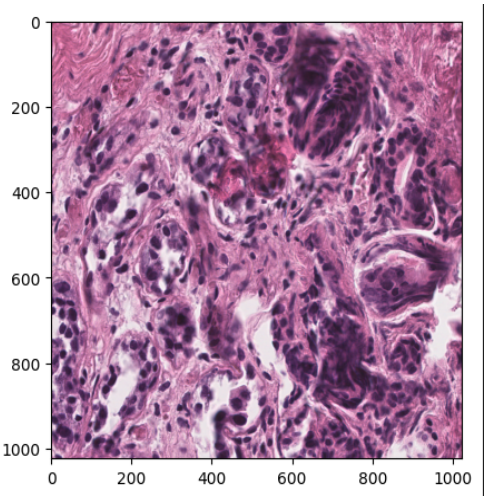


Figure 1.26: Labelled as 'epi' but found in 'stroma' cluster

Figure 1.27: Analysis of patches from BRACS_1918.svs

1.4.1 Problems in the current training method

We observe from the Fig 1.20 that there are many patches having sufficient enough information from multiple classes, but each patch is assigned a single label. This hard-labelling disables us to clearly identify the actual class of the patch due to the presence of information from multiple classes. To address this problem, some of the possible solutions are described in section 1.4.1

Possible Solutions to overcome this problem

1. We can assign multilabels to each patch. For example, if a patch consists of epithetial and stroma cells, we can assign the patch a 6 dimensional vector as label, where the only the elements corresponding to epithelial and stroma class will be 1 and others 0. Now, to create such dataset, it will be an exhausting process if we look at each patch and label it accordingly. To address this, we can use the existing data of 50 WSIs, where we have patches have only one label. We can apply K-means clustering to all the patches from a single class and assign each cluster multiple labels along with the current label. This process can be repeated iteratively to get better clusters and hence better dataset. Once we have such dataset, we can then finally train a classification model which outputs multi-label predictions. We can use this predictions to generate heatmaps where high value represents the high contribution of the particular pixel to one particular class. This information then can be used to generate segmentation masks.

2. Another approach can be to use multiple binary classification models for each class. This model would take a patch from a WSI as an input and predict whether the particular class is whether present or not in the patch. If it is present in the patch, we can use heatmaps to see which region in the patch is contributing the presence of that particular class. The same patch can be passed through all the models corresponding to all the classes and the heatmaps which we get from it can be used to generate segmentation masks. Now, only care that has to be taken while training these models is we have to select patches which consists of pure information from only 1 single class. For example, if we are to train a model to detect whether epithelial cells are present or not in a patch, we have to use dataset which consists of patches which contain only epithelial cells and no other class information.

Chapter 2

Opacity Detection in Chest X-rays using Supervised Contrastive learning

2.1 Internship at Qure.ai

Aim: To train a vanilla classification model on 1.2 million+ Chest X-rays using supervised training to identify the presence of Opacity in an X-ray using conventional supervised training and 'Supervised Contrastive Learning'

2.1.1 Dataset

Total number of Chest X-rays = 1,240,481
All X-rays are 1440x1440 grayscale images resized to 512x512
Labels = Opacity labels (binary) [20.8% labels are positive(1) while 79.2% labels are negative(0)]

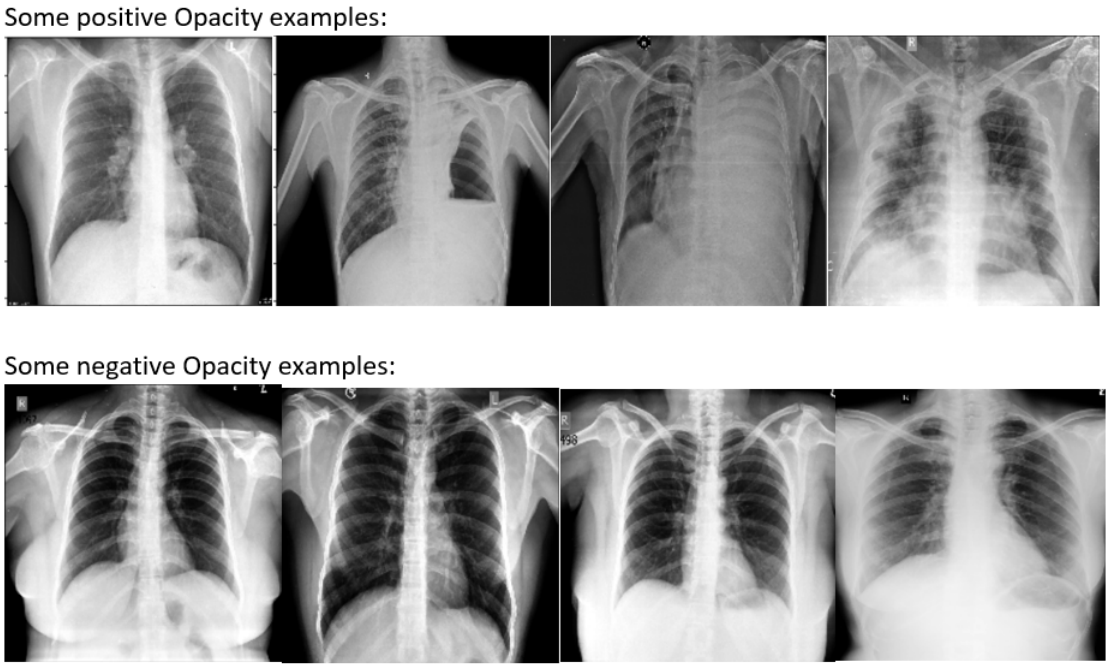


Figure 2.1: Positive and Negative Opacity examples in the Dataset

Train and Validation Data:
Train_validation split = 80% - 20% respectively
Number of Train images = 992,384
Number of validation images = 248,097
Test Data:
All X-rays were resized to the shape 512x512
Number of Images = 286,800
Labels = Opacity labels (binary) [21.5% labels are positive(1) while 78.5% labels are negative(0)]

2.1.2 Baseline Training

Model Architecture:
Resnet50 → 2048 feature vector → Two-layered Projection head → 256 feature vector → Binary classifier with BCE_Loss

Training Specifications: The class imbalance was maintained in both the training and validation dataset as per the full original dataset [20.8% labels are positive(1) while 79.2% labels are negative(0)].
To deal with class-imbalance, resampling strategy was used. During an epoch, in a batch, half of the samples were randomly sampled from positive labels with replacement and half of the samples were sampled from the negative labels iterating over them only once. The model was trained for 52 epochs with a batch size of 20, each batch containing half positive and half negative samples. Adam optimizer was used to minimize the BCE loss with learning rate of 0.005 and scheduler as Reduce LR on plateau (monitoring validation AUC score)

2.1.3 Model training using Supervised Contrastive Learning

The same dataset used in the training of the baseline model was used in this method with the same train and validation data.

Model Architecture:
Resnet50 → 2048 feature vector → Two-layered Projection head → 256 feature vector → Binary classifier with BCE_Loss

Representation learning: In a batch of 16 images of size 512x512, half images were sampled from positive labels with replacement and half from negative labels without replacement. The transformations of RandomCrop(size=390x390), RandomAffine, Random horizontal flip and GaussianBlur were applied on each image twice to get two augmentations for it and a batch 32 images was created. The Supervised Contrastive loss (Eqn 1.2) was optimized using the SGD optimizer with learning rate of 0.005 and scheduler as Reduce LR on plateau (monitoring validation AUC score). The model was trained for 32 epochs only due to low computational power availibilty.

Supervised training: After the representation learning part, the Resnet50 (backbone) weights are frozen and the previous projection head is discarded and a new projection head added with the same architecture as in the baseline (making the SupCon and Baseline architecture identical). And at the end a linear classifier is added with BCE_Loss. Train and validation datasets are same as in the Baseline training method. Here, resampling strategy is used again to handle the class imbalance. The projection head and the classifier is trained for a max of 2-3 epochs since very less no. of parameters have to be trained and we have a very large database.

2.1.4 Results and Observations

The below table (Fig 2.2) summarizes the results obtained for different implementations for the same task of Opacity classification in Chest X-rays.

Model	Tags	Train AUC	VaL AUC	Test AUC	Threshold (test set)	Sensitivity (test set)	Specificity (test set)
BYOL (Resnet50)	Opacity	0.712	NA	NA	NA	NA	NA
Baseline (Resnet50)	Opacity	0.858	0.861	0.8011	0.592	0.724	0.724
SupCon (Resnet50)	Opacity	0.904	0.902	0.837	0.544	0.759	0.759
*Existing Model - Ensemble (Ashish)	Opacity	NA	NA	**0.858**	0.72	**0.76**	**0.79**

* New and cleaner version of Train set was used

Figure 2.2: Comparison of different implementations

Observations:

1. We see a very high jump in the Train, VaL and Test AUCs in the SupCon model as compared to the Baseline model given that both models have same architecture. This concludes that the SupCon method make the model learn a really good set of representations only for the Opacity classification in turn leading to a better result in the classification task.

2. Even with a simple ResNet50 model, the Supervised Contrastive method results are close to the existing results which uses Advanced architectures and ensemble of models like DeepLabv3+ResNext, EfficientNetB7 and updated version of the training dataset.

Chapter 3

Cell Detection using Cell-Tissue Interaction

3.1 Introduction

Cell detection is one of the most important tasks in Computational Pathology. The accurate identification and localization of cells within tissue samples are crucial for quantifying various cellular features, characterizing tissue architecture, and assessing disease progression. Cell detection acts as a cornerstone for a wide range of pathological investigations, from cancer diagnosis and grading to the evaluation of immune responses and the study of infectious diseases.

Pathologists frequently employ a dual approach, zooming out to comprehend tissue-level structures and zooming in to classify cells based on their morphology and the contextual information surrounding them. However, certain deep-learning approaches for cell detection solely focus on utilizing small Field-of-View (FoV) patches to detect and classify cells, neglecting the consideration of the tissue-level structure.

The paper [4] proposes a novel deep-learning approach for cell detection that incorporates the cell-tissue relationship, aiming to mimic the behaviour of pathologists. Additionally, the authors introduce a new dataset named OCELOT in [4], which facilitates the study of the cell-tissue relationship by providing overlapping cell and tissue annotations on images obtained from various organs.

With the objective of advancing the development of robust cell detection methods that harness the power of cell-tissue interaction, the authors of [4] presented an updated version of the OCELOT dataset [4] and organized the OCELOT 2023: Cell Detection from Cell-Tissue Interaction challenge at MICCAI 2023[5]. The challenge aimed to encourage research and exploration on leveraging cell-tissue relationships to enhance cell detection methodologies.

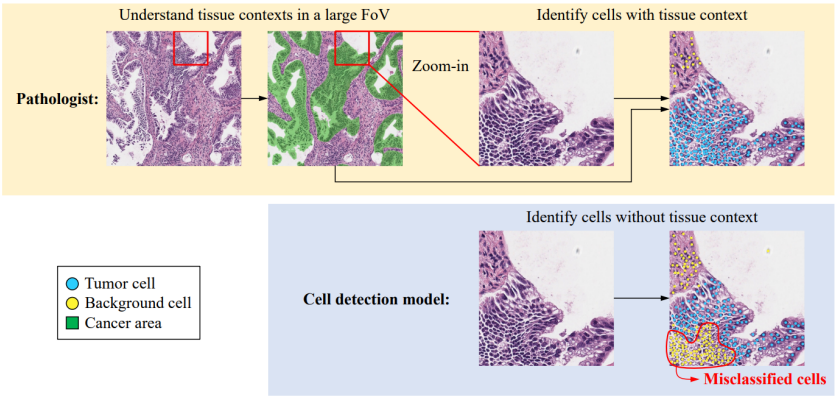


Figure 3.1: Behavior of pathologists and cell detection models [4]

3.2 The OCELOT Challenge 2023

The Dataset encompasses a diverse collection of field-of-view (FoV) patches, including both small and large patches, extracted from digitally scanned whole slide images (WSIs). Notably, these patches exhibit overlapping regions, offering a comprehensive representation of the tissue architecture. The small FoV patches are accompanied by detailed cell annotations, while the large FoV patches provide corresponding tissue annotations. The WSIs within the dataset were sourced from the widely accessible TCGA database[19] and were initially stained using the H&E method before being scanned using an Aperio scanner.
Each sample of the OCELOT dataset is composed of six components,

$$\mathcal{D} = \left\{ \left(x_s, y_s^c, x_l, y_l^t, c_x, c_y \right)_i \right\}_{i=1}^{N} \tag{3.1}$$

where x_s, x_l are the small and large FoV patches extracted from the WSI, y_s^c, y_l^t refer to the corresponding cell and tissue annotations, respectively, and c_x, c_y are the relative coordinates of the center of x_s within x_l. The below figure shows the visualization of a sample.

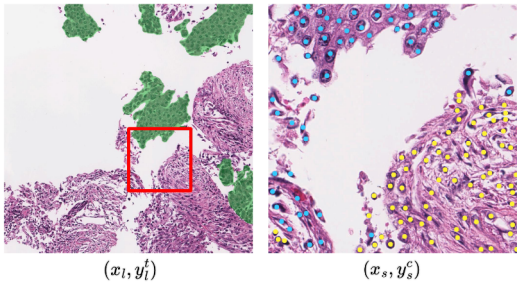


Figure 3.2: A sample from the OCELOT 2023 Dataset [4]. Each sample of the dataset consists of two input patches and the corresponding annotations. Left shows the large FoV patch x_l with tissue segmentation annotation $y_{l'}^t$, where green denotes the cancer area. Right shows the small FoV patch x_s with cell point annotation y_s^c, where blue and yellow dots denote tumor and background cells, respectively. The red box indicates the size and location of the x_s with respect to the x_l.

3.3 Dataset Details

The Dataset comprises of train, validation, and test subsets, consisting of 400, 137, and 130 patch pairs, respectively. Each patch pair within the dataset consists of two components: a tissue patch (large Field-of-View or FoV) with dimensions of 4096x4096 pixels and a cellular patch (small FoV) with dimensions of 1024x1024 pixels. Notably, the cellular patch is entirely contained within the tissue patch, allowing for the complete overlap of cellular and tissue information. To ensure consistency and comparability, the tissue patches are uniformly resized to dimensions of 1024x1024 pixels.
Each cellular patch is accompanied by an annotation file in CSV format. The CSV file contains coordinates of individual cells within the patch, along with their corresponding class labels, which can be either Background cell (BC) or Tumor cell (TC). On the other hand, the tissue patches are annotated using segmentation masks. The annotation for a tissue patch is represented as a segmentation mask, where each pixel is assigned to one of three classes: Background, Cancer area, or Unknown area.

3.4 Evaluation metric

This challenge uses the mean F1 (**mF1**) score as a primary metric which is the average of the F1 score of all cell classes. The F1 score is a commonly used metric for cell detection that considers precision and sensitivity simultaneously. For each cell class, the F1 score is computed by the following equation:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3.2}$$

where $Precision = \frac{TP}{TP+FP}$ and $Recall = \frac{TP}{TP+FN}$
Here, TP, FP and FN denote True Positive, False Positive, and False Negative detections, respectively.

3.4.1 Hit Criterion for cell detection

To determine the TP, FP, and FN, the following process is followed per cell class.

1. Retrieve cell predictions and ground-truth cells from a certain class.
2. Sort cell predictions by their confidence score.
3. Starting from a cell prediction with the highest confidence score, check whether any ground-truth cell is within a valid distance ($\sim$ 15 pixels, $\sim$ 3 um) from the cell prediction.
   (a) If there is no ground-truth cell within a valid distance, the cell prediction is counted as an FP
   (b) If there are one or more ground-truth cells within a valid distance, the cell prediction is counted as a TP. The nearest ground-truth cell is matched with the cell prediction and not considered for further matching.
4. Go back to Step 3 until the cell prediction with the lowest confidence score is reached.

5. The remaining ground-truth cells that are not matched with any cell prediction are counted as FN.

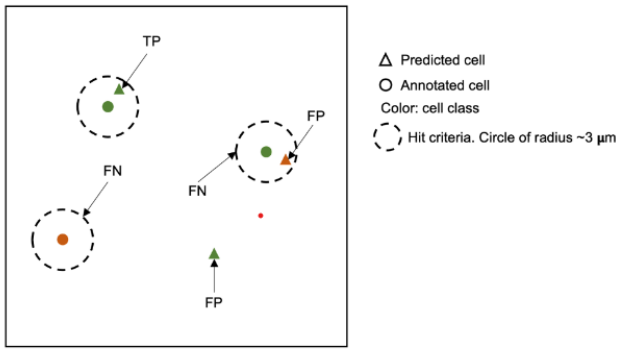


Figure 3.3: Hit criteria for cell detection [5]

3.5 Methodology

We employed various training and inference algorithms and selected the most optimal approach based on rigorous evaluation. Herein, we present the different strategies that were explored and analyzed to identify the best-performing algorithm.

3.5.1 YOLOv8 objection detection model

Given the similarity between detecting cell locations and object detection, we initially employed the YOLOv8[20] (current state-of-the-art object detection model) for detecting the bounding boxes corresponding to cells. Our approach commenced by focusing solely on the cellular patches, where we annotated bounding boxes measuring 30x30 pixels, which aligns with the typical size of a cell within the patch. These bounding boxes were centered around the coordinates provided in the ground truth CSV labels.

Training Details

1. As the organizers provided us with access only to the training data, which consisted of 400 patch pairs, we conducted a random split of the dataset into an 8:1:1 ratio for training, validation, and testing, respectively. This particular split was consistently utilized for all subsequent analyses and purposes throughout the study.

2. Since, the train data was very limited, the following data augmentations were used:

   (a) Random Horizontal and Vertical Flip

   (b) 90^o rotation, clockwise or counter-clockwise

   (c) Rotation (between $-15^o$and 15^o)

   (d) Shear ($\pm 15^o$ Vertical, $\pm 15^o$ Horizontal)

3. For cell detection, we employed a YOLOv8 model with a medium size, encompassing approximately 25 million parameters. To facilitate efficient training, the input

images were resized to dimensions of 736x736 pixels. The model was trained for 100 epochs, and the best-performing model, as determined by its performance on the validation set, was saved for further evaluation and predictions.

Results

To obtain optimal results during the prediction phase, we used a confidence threshold (conf) of 0.2 and an intersection over union (IOU) threshold of 0.5 while predicting with the trained YOLOv8[20]. These threshold values were determined through experimentation and resulted in the most accurate predictions for cell detection.

	Pre (BC)	**Re (BC)**	**F1 (BC)**	**Pre (TC)**	**Re (TC)**	**F1 (TC)**	**mF1**
Train	0.676	0.8247	0.743	0.8784	0.7265	0.7952	0.769
Valid	0.5423	0.6083	0.5734	0.8067	0.6392	0.7132	**0.643**
Test	0.5281	0.4933	0.5101	0.7748	0.6566	0.7108	**0.610**

Table 3.1: BC: Background cell, TC: Tumuor Cell, Pre: Precision, Re: Recall

3.5.2 Visualization of the predictions

In the below images, Red boxes indicate Tumor Cells (TC) and the Green boxes indicate (BC)

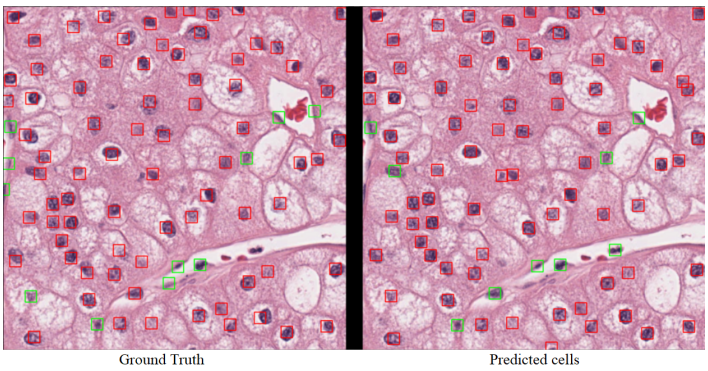


Figure 3.4: YOLOv8 correctly predicting cell locations and their classes

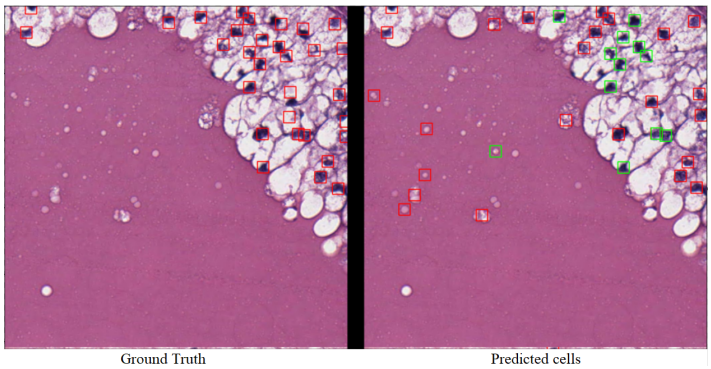


Figure 3.5: YOLOv8 incorrectly predicting cell locations

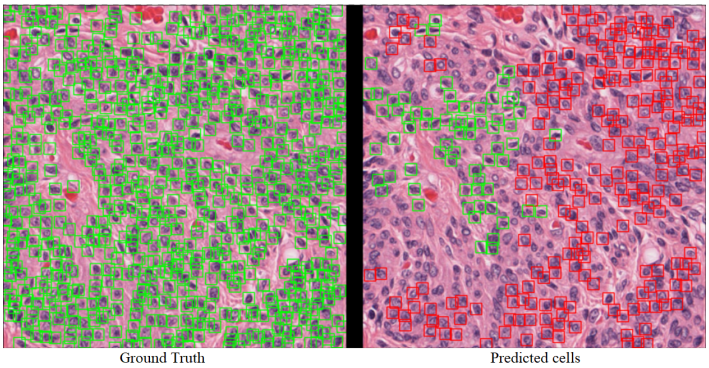


Figure 3.6: YOLOv8 incorrectly predicting the classes of predicted cells

3.5.3 YOLOv8 with a cell-classifier

Upon evaluating the predictions made by the aforementioned YOLOv8 model on both the test and validation sets, it was evident that the model successfully detected the cell locations with a reasonable level of accuracy. However, we noticed several instances in which the classification of cells was inaccurate. To address this issue, we hypothesized that the imbalanced distribution between the number of tumor cells and background cells might have contributed to the poor classification performance.

In light of this, we decided to employ a separate classifier to improve the accuracy of cell classification. The classifier was trained specifically on patches measuring 128x128 pixels, providing sufficient coverage for an entire cell. These patches were extracted from the training dataset and centered around the coordinates provided in the ground truth CSV files. By training the classifier on these patches while sampling them uniformly from both classes, we aimed to mitigate the issues arising from imbalanced class distribution and enhance the accuracy of cell classification, independent of the YOLOv8 model's classifications.

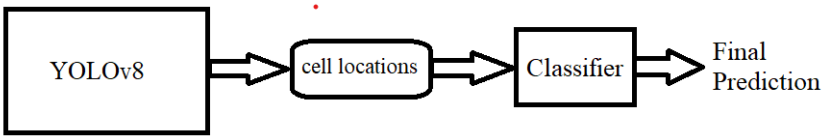


Figure 3.7: Proposed pipeline for YOLOv8

Results

	Pre (BC)	Re (BC)	F1 (BC)	Pre (TC)	Re (TC)	F1 (TC)	mF1
Train	0.7228	0.8255	0.7708	0.8635	0.7539	0.805	0.7879
Valid	0.4857	0.6271	0.5474	0.7912	0.5857	0.6731	**0.610**
Test	0.5151	0.5585	0.5359	0.7891	0.6033	0.6838	**0.609**

Table 3.2: BC: Background cell, TC: Tumuor Cell, Pre: Precision, Re: Recall

Based on the analysis of the obtained results, we made the decision to **discard** the aforementioned model, as no noticeable improvement was observed in comparison to the classification results achieved by the YOLOv8 model. Our analysis led us to believe that in order to achieve accurate cell classification, a broader context, such as the tissue region from which the cell patch is extracted, is required.

To address this challenge, we hypothesized that incorporating information from the tissue segmentation results could potentially enhance the cell classification task. The tissue segmentation results, which also encompass two classes that align with the classes we aim to classify (e.g., background and tumor cells), have the potential to provide valuable contextual information. By leveraging the tissue segmentation results, we anticipate an improvement in the classification accuracy, as it allows for a more comprehensive understanding of the local tissue environment surrounding the cells. We incorporate this in our next model.

3.5.4 YOLOv8 with Tissue segmentation model

Training details of Tissue Segmentation Model

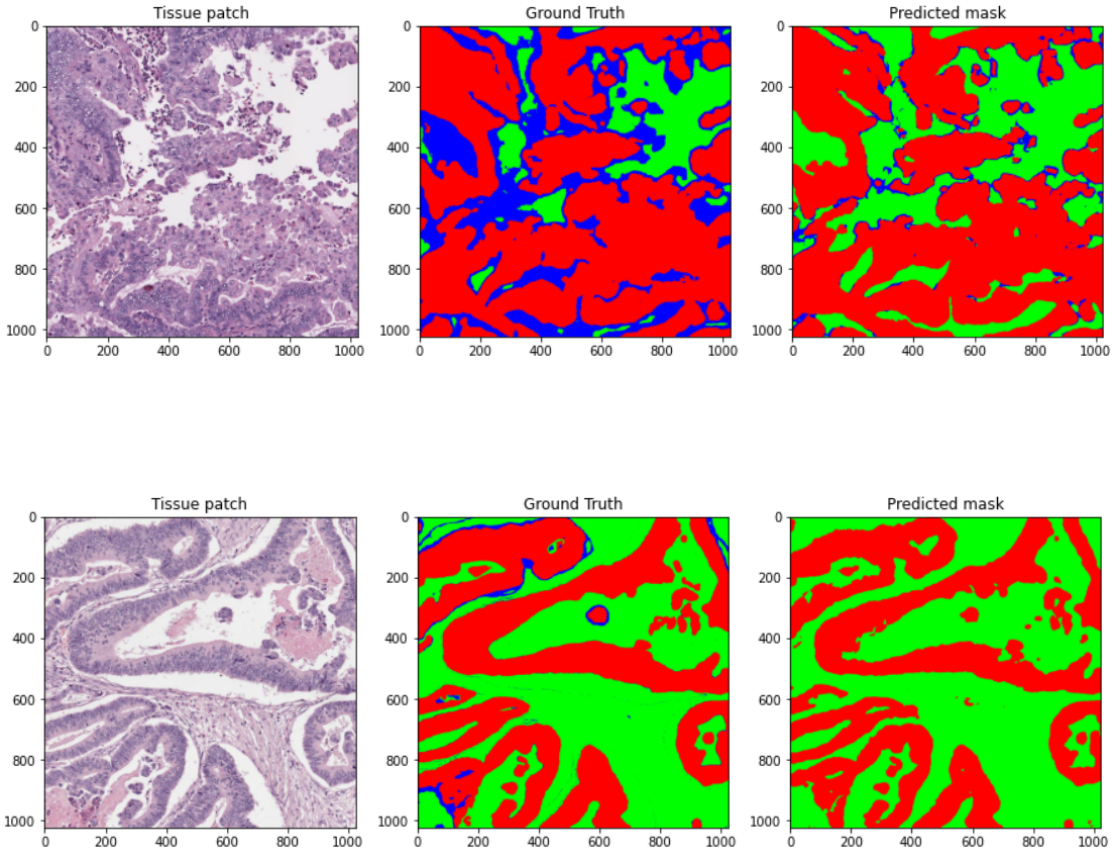
The Tissue Segmentation model implemented in this research study adopts the DeepLabv3+ architecture[21], using a resnet34 encoder[22]. To train the model, tissue patches from the training set were utilized, along with their corresponding ground truth annotations. The ground truth annotations encompassed three distinct classes: Background area, tumor area, and unknown area.

Given the limited availability of training data, various data augmentation techniques were employed to augment the dataset. These augmentations included random horizontal and vertical flips, as well as 90^o rotations (both counterclockwise and clockwise).

During the training process, the Tissue Segmentation model was trained for 200 epochs, utilizing a batch size of 32. The Adam optimizer was employed to optimize the model's parameters. To further optimize the learning process, a Cosine Annealing learning rate scheduler was employed. The model was trained using the cross-entropy loss function, with the objective of minimizing the discrepancy between predicted and ground truth segmentation masks. Ultimately, the model with the lowest validation loss was selected as the optimal model for further analysis and evaluation.

Visualization of predicted tissue segmenation masks

In the below figures, the Green area represents the background area class, the Red area indicates the Tumor region while the blue area indicates unknown class. All the tissue patches below are from the validation set.



3.5.5 Integration of YOLOv8 and Tissue segmentation model

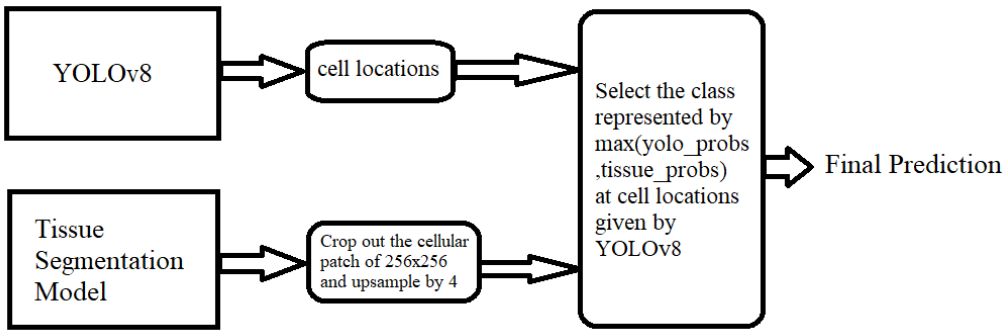


Figure 3.8: Integrating tissue segmentation predictions with YOLOv8 results

In order to integrate the predictions of the Tissue Segmentation model with YOLOv8, a specific process was followed. Initially, the output mask, containing probabilities, generated by the Tissue Segmentation model was utilized. From this output mask, a 256x256 patch was cropped, which corresponded to the cellular patch within the given data point. The metadata of the dataset provided the center coordinates of the cellular patch within the tissue patch. To ensure compatibility, the 256x256 patch was then

upsampled four times using nearest neighbor interpolation, aligning it with the size of the cellular patch.

Subsequently, the cell locations predicted by the YOLOv8 model were incorporated. For the purpose of classification, the class was determined by selecting the maximum probability value between the probabilities obtained from the YOLOv8 model and the tissue segmentation probabilities for the specific cell location. By combining these probabilities, the classification process aimed to leverage the strengths of both models, thereby improving the overall accuracy and reliability of cell classification.

Results

	Pre (BC)	**Re (BC)**	**F1 (BC)**	**Pre (TC)**	**Re (TC)**	**F1 (TC)**	**mF1**
Train	0.6592	0.7041	0.6809	0.7814	0.7164	0.7475	0.7142
Valid	0.6123	0.5441	0.5762	0.7768	0.6712	0.7201	**0.648**
Test	0.6584	0.5133	0.5769	0.7697	0.718	0.743	**0.66**

Table 3.3: BC: Background cell, TC: Tumuor Cell, Pre: Precision, Re: Recall

The integration of the tissue segmentation model's predictions into the classification task has demonstrated its effectiveness, as evidenced by the notable increase in the mean-F1 score on both the validation and test sets. These results unequivocally indicate the significance of considering a broader context when determining the appropriate class for a given cell.

By incorporating the tissue segmentation model's predictions, which encompass information about the surrounding tissue environment, the classification process becomes more comprehensive and accurate. The improved mean-F1 score highlights the value of incorporating contextual information to enhance the understanding and classification of cells.

3.5.6 Cell detection by Cell-only segmentation method

In this section, we explore a new approach for cell detection by reframing the problem as a segmentation task, diverging from the conventional perspective of object detection. Initially, we focus exclusively on utilizing the cellular patches derived from the training set. To facilitate this approach, ground masks are generated with dimensions of 1024x1024x3, where each channel represents a specific class: Tumor cell, background cell, and background region.

To create the ground masks, we adopt a circular representation methodology. Fixed-radius circles, measuring 15 pixels in diameter, are drawn around the cell coordinates provided in the ground truth CSV files. These circles are placed in the respective channel corresponding to the class of the specific cell. This process enables the creation of accurate and comprehensive masks, encompassing the desired cell classes within the cellular patches.

Training Details of the Cell-only segmentation model

We employed the DeepLabv3+ architecture[21] with a 'resnet34' encoder[22] to train a model utilizing the cell patches and ground truth masks derived from annotated CSV files. Due to the limited availability of training data, we applied various data augmentation techniques to augment the dataset. These augmentations included random horizontal/vertical flips, 90-degree rotations (both clockwise and counterclockwise), and ColorJitter transformations. These techniques aimed to enhance the model's ability to generalize and alleviate potential issues related to overfitting.

For training, we utilized the Multilabel Dice loss function[23] in conjunction with an Adam optimizer[15]. To optimize the learning process further, we incorporated a Cosine Annealing learning rate scheduler[24]. The model underwent training for a total of 200 epochs, and the model with the best validation loss was saved for subsequent analysis and evaluation. This comprehensive training process enabled the model to learn and extract meaningful features from the cell patches, enhancing its ability to accurately segment and classify cells in the subsequent stages of the study.

Visualization of the Preicted Heatmaps

In the below figures, the blue colour represents the background area, the Green colour represents the Background cells and the Red colour indicates Tumour cells.

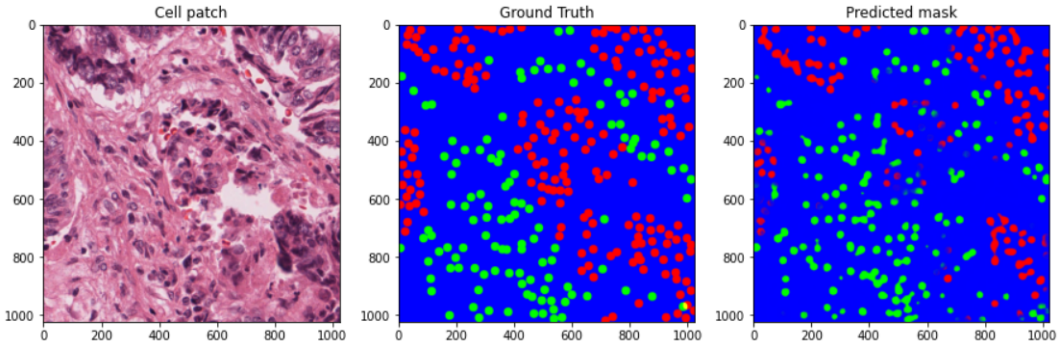


Figure 3.9: An example output of cell segmentation model from validation set

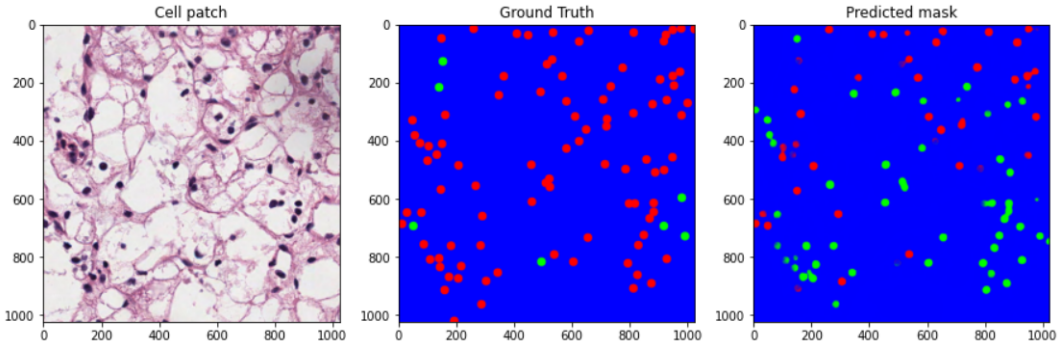


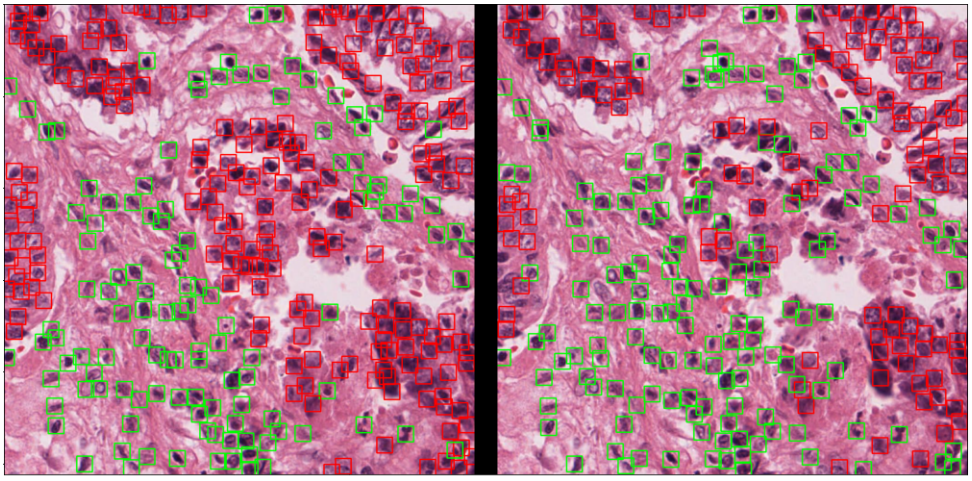
Figure 3.10: An example output of cell segmentation model from validation set

Cell detection using the Probabilty heatmap

After obtaining the probability heatmap from the cell-only segmentation model, we employ a systematic algorithm to extract the precise cell locations and determine their respective classes. The steps involved in this algorithm are as follows:

1. Split the channels of the heatmap as background channel (1024x1024x1) and non-background channels (1024x1024x2)
2. Calculate the detection score by subtracting the background channel from 1.0.
3. Smooth the detection score using a Gaussian filter with a 5x5 kernel and $\sigma = 3$.
4. Find the peaks in the smoothened detection score, considering a minimum distance of 10 pixels between peaks using the library `feature.peak_local_max` from `scikit-image`
5. Compute the maximum values and classes for each spatial location.
6. Filter out peaks where the background score is higher than the maximum value.
7. Extract the scores and classes of the remaining peaks.

An example output of using this algorithm on the heatmap from Fig 3.9 is as follows:



Ground Truth Cell detection using proability heatmap

Results

	Pre (BC)	Re (BC)	F1 (BC)	Pre (TC)	Re (TC)	F1 (TC)	mF1
Train	0.8301	0.8177	0.8238	0.8907	0.8124	0.8498	0.8368
Valid	0.6319	0.5679	0.5982	0.7958	0.719	0.7554	**0.677**
Test	0.6636	0.5785	0.6181	0.7807	0.6735	0.723	**0.67**

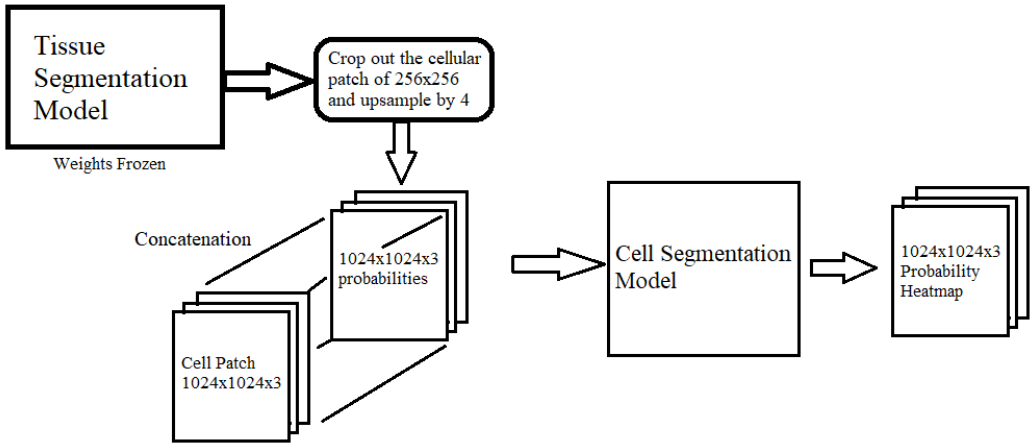
Table 3.4: BC: Background cell, TC: Tumuor Cell, Pre: Precision, Re: Recall

The results obtained clearly demonstrate that the segmentation-based approach significantly outperforms the YOLOv8-based cell detection method, both in terms of validation and test datasets. The margin of improvement observed highlights the efficacy of the segmentation approach in accurately detecting and classifying cells.

While the segmentation-based approach proved successful, it is important to note that it did not incorporate the utilization of tissue patches. In the subsequent section, we aim to further enhance the results by leveraging the inclusion of tissue patches. In the Section 3.5.4, we saw that incorporating tissue segmentation for classification, in conjunction with the YOLOv8 method, yielded even better performance. The promising improvement observed when utilizing tissue segmentation in the previous experiments motivates us to explore its potential for further enhancing the overall scoring metrics.

3.5.7 Cell-Tissue Segmentation Model

The idea behind this model is to utilize the tissue segmentation predictions while training the cell segmentation model. The Figure below represents the pipeline we follow to get the output as a probability heatmap.



We leverage the Tissue Segmentation model trained in Section 3.5.4, utilizing its frozen weights. To create the input for this approach, we construct a multi-channel image with dimensions of 1024x1024x6. This is achieved by concatenating the cell patch with the cropped and upsampled (4 times) tissue segmentation predictions, wherein the cropped portion of the prediction aligns with the cellular region within the corresponding tissue patch.

By combining the cell patch and the augmented tissue segmentation predictions, we create a comprehensive input representation that captures both the cellular and tissue context.

Training Details

We utilized the DeepLabv3+ architecture [21] with a 'resnet34' encoder[22] to train a model that integrates cell patches, tissue segmentation predictions, and ground truth masks obtained from annotated CSV files. To address the limited availability of training

data, we employed various data augmentation techniques on the cell patches. These techniques encompassed random horizontal/vertical flips, 90-degree rotations (both clockwise and counterclockwise), and ColorJitter transformations.

During the training process, we employed the Multilabel Dice loss function [23] in conjunction with the Adam optimizer[15]. To further enhance the learning process, we integrated a Cosine Annealing learning rate scheduler [24]. The model was trained for 200 epochs, and the version with the lowest validation loss was selected as the final model for subsequent analysis and evaluation.

Results

	Pre (BC)	**Re (BC)**	**F1 (BC)**	**Pre (TC)**	**Re (TC)**	**F1 (TC)**	**mF1**
Train	0.8301	0.8177	0.8238	0.8907	0.8124	0.8498	0.8368
Valid	0.5995	0.6356	0.617	0.7874	0.7184	0.7513	**0.684**
Test	0.6592	0.678	0.6685	0.8097	0.7196	0.762	**0.715**

Table 3.5: BC: Background cell, TC: Tumuor Cell, Pre: Precision, Re: Recall

3.6 Results and discussion

In this section we compare the validation and test results of all the above models in the following table.

Model/Algorithm	**Validation (mF1 score)**	**Test (mF1 score)**
YOLOv8	0.643	0.61
YOLOv8+Classifier	0.610	0.609
YOLOv8+Tissue segmentation for classification	0.648	0.66
Cell Segmentation	0.677	0.67
Cell-Tissue Segmentation	**0.684**	**0.715**

Chapter 4

Development of a Quality Control tool for WSIs using Deep learning

4.1 Introduction

Histopathology, the microscopic examination of tissue specimens, plays a pivotal role in the diagnosis, prognosis, and treatment planning of various diseases, including cancer. As the demand for accurate and efficient diagnostic tools continues to grow, digital pathology has emerged as a transformative field, enabling the digitization of histological slides into Whole Slide Images (WSIs). These digital representations allow pathologists to remotely access, share, and analyze tissue samples with unprecedented convenience and collaboration potential. However, the successful adoption of digital pathology critically relies on the development and implementation of robust quality control mechanisms, ensuring the reliability and accuracy of WSIs.

Quality control in histopathology encompasses a broad range of challenges, including staining variations, tissue artifacts, scanning artifacts, and image artifacts, among others. These factors can introduce significant variability and may compromise the accuracy and reproducibility of diagnostic interpretations. Manual inspection by expert pathologists remains the gold standard for quality control, but it is time-consuming and subjective. Therefore, there is an urgent need for automated, objective, and efficient quality control techniques that can ensure the integrity and consistency of WSIs, thereby improving diagnostic accuracy and patient care. In this research paper, we present a novel technique that focuses on training segmentation models specifically tailored for histopathology images. Our approach addresses the inherent challenges associated with histopathology image analysis, particularly the accurate identification of distinct tissue components and the detection of various artifacts. To tackle these challenges, we have devised four segmentation models: blur level segmentation, tissue segmentation, tissue fold segmentation and pen marker segmentation. A significant contribution of our study lies in the integration of domain knowledge derived from the HistoROI model. This domain knowledge serves as a valuable resource for optimizing the data sampling process during training, ultimately leading to the development of more precise and robust segmentation models. By harnessing the unique characteristics of histopathology images and leveraging our innovative training technique, we strive to improve the overall

accuracy and reliability of histopathology image analysis.

4.2 Review of Literature

4.2.1 HistoQC: Quality Control Tool for Digital Pathology Slides

Introduction

Various types of artifacts can be introduced in the final Whole Slide Image (WSI) as a result of small, unavoidable errors during the slide preparation and digitization process. Manual review of glass and digital slides is laborious and subject to high variability among different pathologists. To address these challenges, the authors of [10] have developed a tool that employs a reproducible automated approach to precisely localize artifacts. This tool aims to identify slides that need to be reproduced or regions that should be avoided during computational analysis.

Methods

The HistoQC tool[10] is executed by providing a configuration file containing user-defined parameters. These parameters specify the modules to be run, their order of execution, the level of the WSI for image extraction, various kernel sizes and thresholds for different modules, and other relevant parameters. The tool is implemented using a python-based pipeline, which executes all the modules mentioned in the configuration file. Upon completion, the tool generates several output images, including a thumbnail of the WSI, a mask indicating the useful region within the WSI, a mask highlighting blurry locations, a mask identifying pen markings, and more. HistoQC also offers an interactive graphical user interface that presents the user with identified regions free of artifacts along with associated metrics. This interface allows for real-time visualization and filtering, which greatly facilitates the detection of artifacts

Results and Discussion

To validate the results provided by the HistoQC tool, two pathologists experienced in digital pathology were enlisted to grade each of the output masks generated by HistoQC as either acceptable or not acceptable. Acceptance required a minimum of 85% area overlap between the pathologists' visual evaluation and HistoQC's computational evaluation of artifact-free tissue. The overall agreement between HistoQC and the expert pathologists was found to be 95% (477 out of 500), indicating a high level of concordance between the tool and human experts.

4.3 Methodology

In this study, our main focus was on accurately identifying and categorizing artifacts present in medical images. To accomplish this objective, we developed four distinct models for segmenting artifacts, each specifically designed to address a specific type: blur level, tissue fold, pen marker, and tissue segmentation.

4.3.1 Pen Marker Segmentation Model

To ensure the availability of annotated samples for training the pen marker segmentation models, a specific dataset was used in this experiment. The dataset consisted of images that were annotated at 0.625X magnification and included various colored pen markers, such as black, red, green, yellow, and more. The inclusion of a diverse range of pen marker colors commonly found in pathology aimed to make the dataset representative of real-world scenarios.

To train the pen marker segmentation model, a data sampling strategy was implemented. This strategy involved randomly selecting a mask from the dataset and then randomly sampling a positive pixel, which represents a pixel with a pen marker, within that mask. A patch of size 512x512 was extracted around the randomly sampled pixel, serving as the input to the model during the training process. By employing this approach, the model was exposed to a diverse set of examples during training, mitigating the risk of overfitting and improving its generalization performance.

During training, a combination of cross-entropy, focal, and dice loss functions were utilized to optimize the model. This optimization strategy aimed to enhance the model's ability to accurately segment pen markers in different image contexts. Fig 4.1 in the research paper provides a visual representation of an annotated image from the dataset, showcasing the successful segmentation of pen markers.

Figure 4.1: Example of Pen annotation

4.3.2 Tissue Folds Segmentation Model

For the training of the Tissue folds segmentation model, a dataset comprising 250 patches from the BRIGHT[13] dataset was used. The images in the dataset were annotated specifically for tissue folds at a magnification level of 5x to ensure accurate capture of these features. The same training strategy employed for pen marker segmentation (Section 4.3.1) was also utilized for tissue segmentation to promote generalization performance of the model.

The below Figure 4.2, an example of a tissue folds image along with its corresponding segmentation mask is presented.

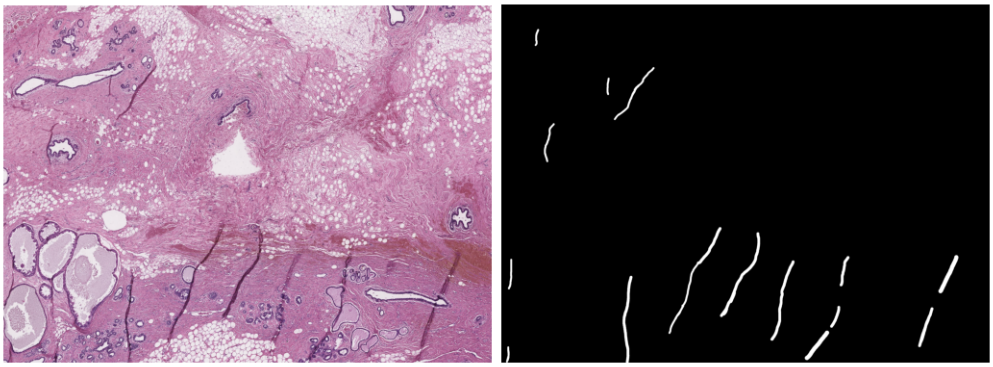
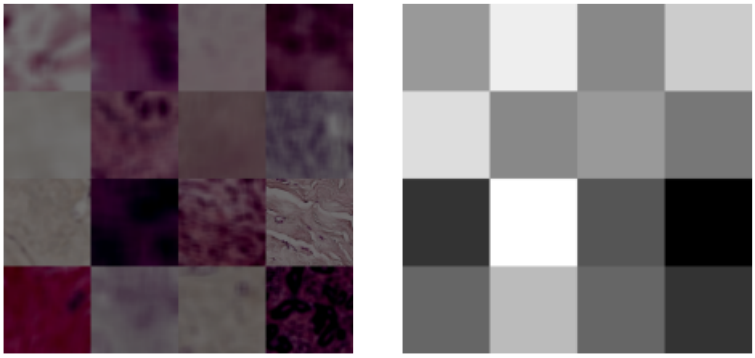


Figure 4.2: Example of a Tissue Fold

4.3.3 Blur level Segmentation Model

Detecting blur levels in whole slide images (WSI) is challenging due to texture variations across different regions. Traditional laplacian-based methods struggle with consistent performance, especially when faced with diverse textures like normal stroma versus cellular regions. To address this, we combined a patch mining approach with the HistoROI model. By utilizing patch mining, the segmentation model learns typical texture patterns for specific regions in the WSI, overcoming the limitations of laplacian-based methods. The HistoROI model helps differentiate between regions, enhancing accuracy in blur detection. For training the segmentation model, we applied synthetic blur to patches identified as foreground by HistoROI. Multiple blur levels were used, and the model was trained on 5x magnification patches, utilizing the boxblur function of the PIL library to generate synthetic blur.



(a) (b)

Figure 4.3: Example of 4.3(a) Blur levels and 4.3(b) masks

4.3.4 Tissue Segmentation model

We focused on tissue segmentation in whole slide images (WSI) at a 2.5x magnification level, striking a balance between capturing tissue structures and computational efficiency. Using the HistoROI model and the cut-paste method, we extracted relevant patches for tissue detection and incorporated them into our training dataset. One

challenge in tissue detection is differentiating between background and adipose tissue. To address this, we treated adipose tissue as a separate class, enabling more accurate differentiation. To enhance the model's robustness in identifying artifacts and pen markers, we applied heavy color jitter augmentation to background patches, creating diverse training data. This approach improved the model's ability to accurately detect tissue regions and artifacts, enhancing overall performance.

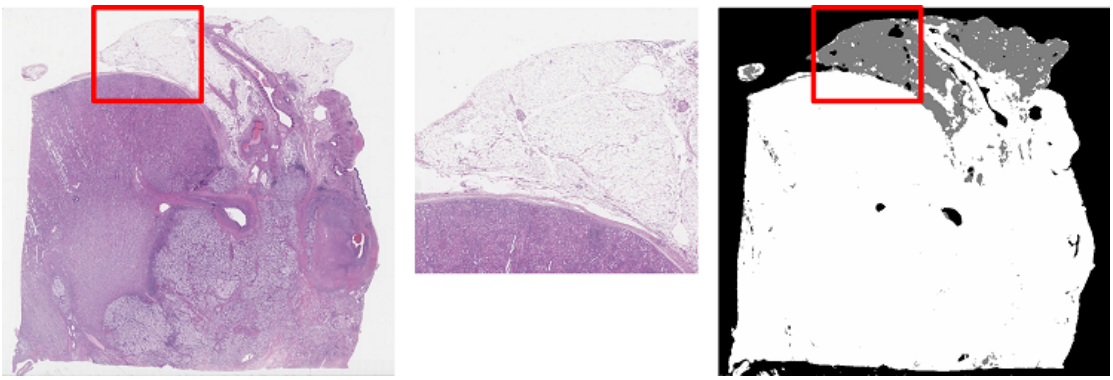


Figure 4.4: Example of Tissue vs background vs Adipose in a WSI

Cut-Paste Method

Training accurate segmentation models for whole slide images (WSI) can be challenging due to limited annotated data and imbalanced data distribution. To address these challenges, we employed a cut-paste method for training segmentation networks. This involved cutting patches from WSIs and pasting them into smaller fixed-size patches for training, ensuring representative samples in each training batch and mitigating skewed data distribution. Additionally, we integrated the HistoROI patch classification model to enhance segmentation accuracy. By utilizing the HistoROI model, we selectively chose patches or regions from specific classes, such as epithelial, stroma, lymphocytes, adipose, miscellaneous, or artifact, during training, ensuring a balanced representation of different tissue regions.
The HistoROI model, trained using a deep neural network on patch-level data, demonstrates remarkable accuracy in patch classification. Through the integration of the cut-paste method and the HistoROI model, we successfully overcome the obstacles presented by limited annotated data and imbalanced data distribution. This combined approach empowers us to construct segmentation models for WSI analysis that are both robust and highly accurate.

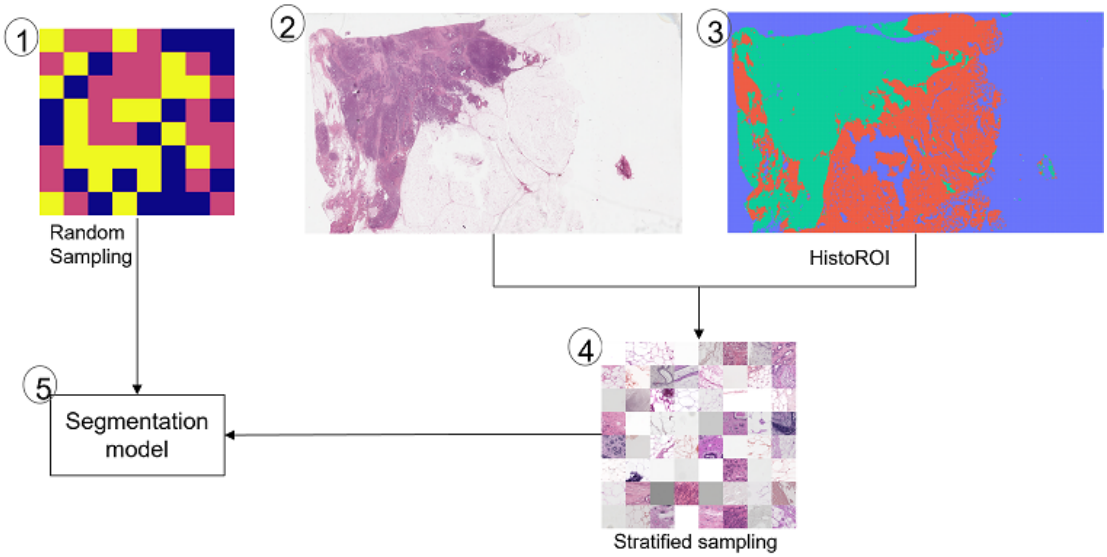


Figure 4.5: Illustration of cut-paste method. (1) Random Sampling: In a big mask randomly select the labels for patch size, (3) Output of HistoROI for WSI in (2), (4) stratified sampling is applied using HistoROI output and the patches from WSI is selected, (5) Randomly generated mask label and stratified sampling are fed to the segmentation model for training.

4.3.5 Model Architectures

The segmentation models considered in this research are all based on the UNet++ architecture, a widely recognized framework for image segmentation. For the pen marker segmentation model, we employed ResNet34 as the backbone, while EfficientNet-b0 was chosen as the backbone for the blur segmentation, tissue fold segmentation, and tissue segmentation models. These backbones were pre-trained on the Imagenet dataset, ensuring their ability to extract relevant features from the input images effectively. To achieve efficient execution, especially at higher magnifications, we opted for the lightweight EfficientNet-b0 backbone for the latter three models.
By employing the UNet++ architecture, pre-trained backbones, and the convenience of the segmentation-models-pytorch library, we established a robust and efficient framework for training our segmentation models. These models serve as integral components of our comprehensive quality control framework for whole slide images, providing accurate and reliable segmentation results.

4.4 Experiments and Results

This section presents the experiments performed to validate our approaches for quality control and nuclei density prediction. We assessed the accuracy of our final usable masks by comparing them to the functional masks generated by the HistoQC tool. The dice score was employed to measure the similarity and overlap between the two sets of masks, enabling us to evaluate the effectiveness of our segmentation models in generating high-quality masks. Furthermore, we evaluated the performance of our blur level prediction

model on publicly available datasets, namely FocusPath[25] and TCGA@Focus[26].

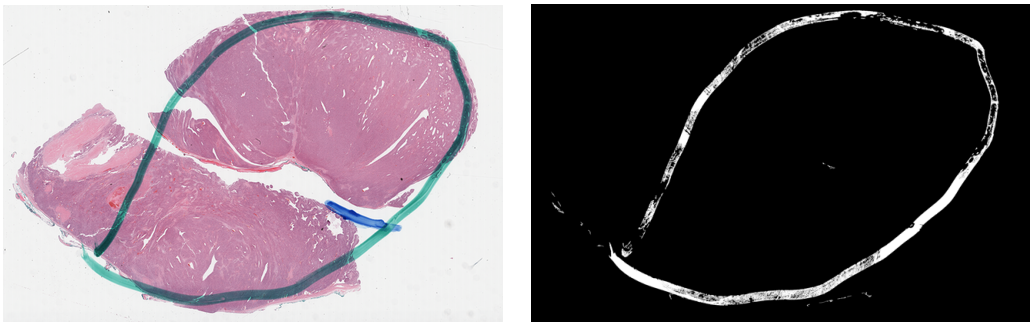
4.4.1 Blur Level Segmentation Model Performance

The research paper discusses two datasets used in the experiments. The FocusPath dataset consists of diverse Whole Slide Image (WSI) scans captured at different focus levels, exhibiting varying degrees of blur. It includes 864 image patches of size 1024x1024 with different levels of blur. Ground truth scores indicating the focus level of each image patch are provided, making this dataset valuable for evaluating focus quality in digital pathology and microscopy images. The TCGA@Focus dataset, obtained from The Cancer Genome Atlas repository, consists of 1000 whole slide images representing 52 organ types. Each region of interest in the images is annotated as "in-focus" or "out-focus" and assigned binary ground truth scores. This dataset contains 14,371 image patches, enabling a comprehensive evaluation of the blur level prediction model across diverse tissue textures and color information.

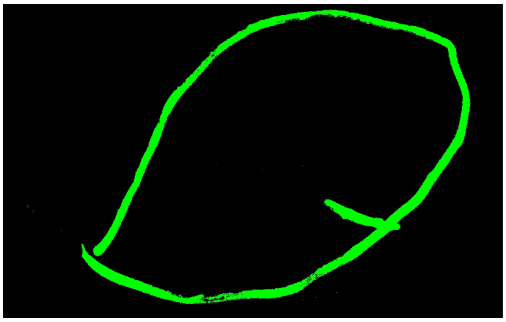
As the datasets were initially intended for classification tasks, the research paper describes a method to estimate the blur level as a classification probability. The approach involves calculating the mean of the predicted segmentation mask, summarizing the blur level information from segmented regions to obtain an overall blur level value. This enables the transformation of the output of the segmentation model into a format interpretable as a classification result. The experiments conducted on the datasets demonstrated the model's performance in predicting blur levels, achieving a Receiver Operating Characteristic - Area Under the Curve (ROC-AUC) of 0.883 on the FocusPath dataset and 0.786 on the TCGA@Focus dataset. These results highlight the effectiveness of the model in predicting blur levels and its potential for application in blur assessment tasks.

4.4.2 WSI Profiler

In this experiment, we proposed a methodology to generate a final usable mask by integrating the outputs of multiple models. Handling the large size of the Whole Slide Images (WSIs) posed a challenge, as it was impractical to load the entire slides into the system, even at 5x resolution. To overcome this limitation, we employed a strategy where the WSIs were divided into smaller sections, processed individually, and then combined to create a mask for the entire slide image.
The workflow consisted of sequentially applying various models to the WSIs. The first step involved tissue detection, followed by blur-level detection, tissue fold detection, and pen detection. These models were executed on a comprehensive dataset comprising 11,529 whole slide images sourced from the Cancer Genome Atlas (TCGA) repository. Additionally, we conducted HistoQC analysis on all 11,529 images to serve as a basis for comparison and evaluation purposes.
The visual comparisons between the output of HistoQC and our WSI profiler are depicted in the accompanying figures. These figures provide a visual representation of the similarities and differences in the results obtained from both methods.



(a) (b)



(c)

Figure 4.6: Pen model prediction 4.6(a) shows the thumbnail of the image, 4.6(b) shows the histoQC pen marking output and 4.6(c) shows the wsi profiler pen marking segmentation model output

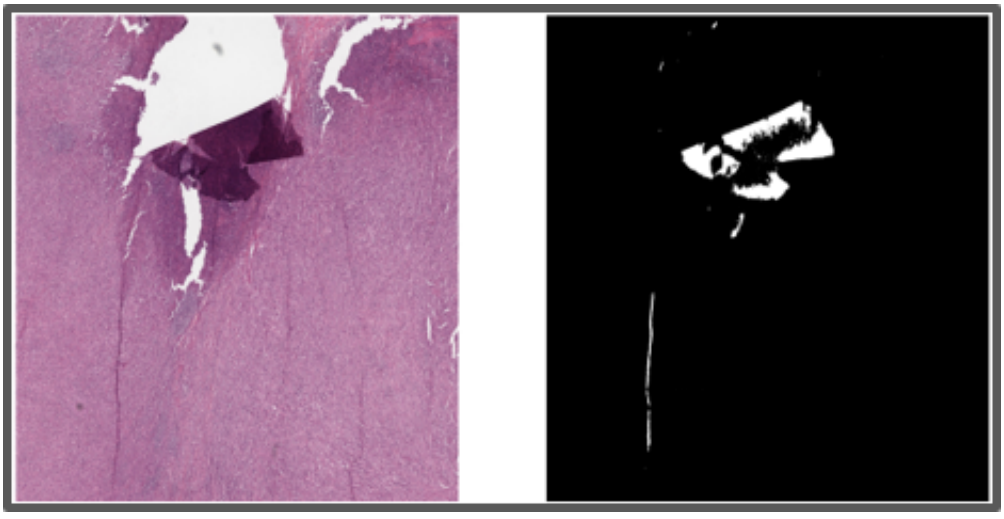
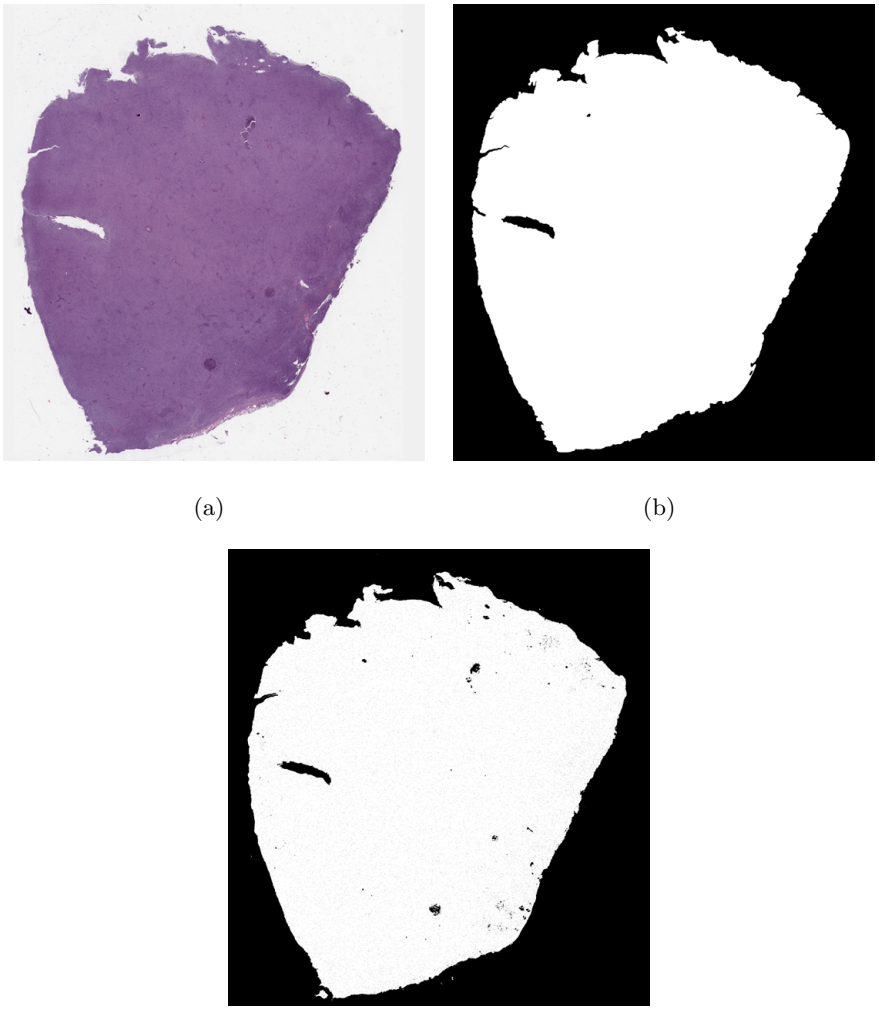


Figure 4.7: Tissue folds model prediction



(a) (b)

(c)

Figure 4.8: Final useful mask prediction 4.8(a) shows the thumbnail of the image, 4.8(b) shows the histoQC output and 4.8(c) shows the our wsi profiler output

Additionally, in order to assess the performance of our WSI profiler model, we employed the Dice score metric to compare the generated masks with those produced by HistoQC. The Dice score quantifies the degree of overlap between two binary masks, serving as a measure of their similarity. The distribution of Dice scores is illustrated in Figure 4.9. The results indicate a high level of agreement between HistoQC and our WSI profiler model for the majority of WSIs. Specifically, out of the 11,529 WSIs evaluated, approximately 74% (8,847 WSIs) exhibited a Dice score exceeding 0.7, signifying a substantial level of concordance between the two methods.

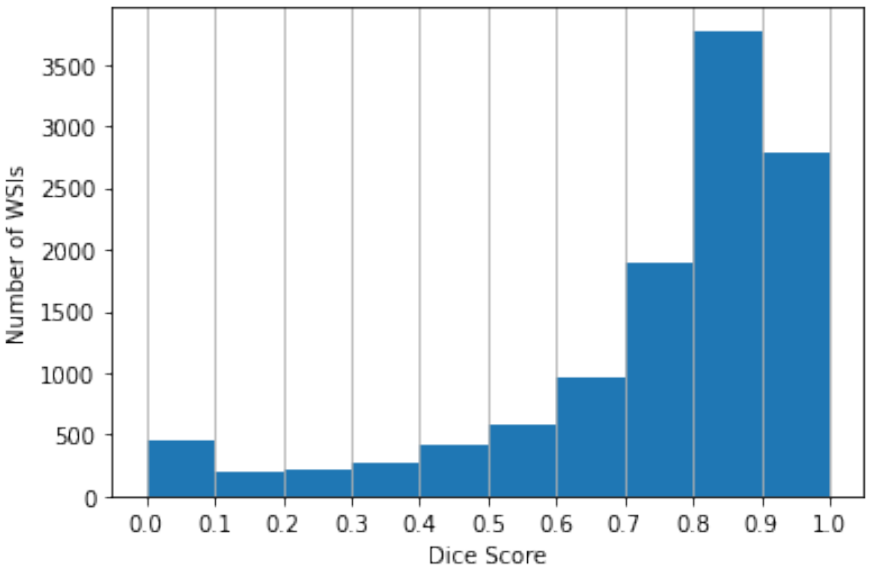


Figure 4.9: Histogram of Dice score between HistoQC and our WSI profiler models for TCGA WSIs

Chapter 5

Conclusion and Future Work

5.1 Conclusions

5.1.1 HistoROI: Histopathology specific preprocessing

In this study, we used the human-in-the-loop active learning paradigm to prepare a patch-level dataset for classifying pathology-relevant region of interests (ROIs). We trained a 6-class classification model HistoROI using the prepared data to classify the patches into one of the following - epithelial, stroma, lymphocytes, artifacts, miscellaneous and adipose. We investigated the performance of HistoROI on the predictions of atrifacts by comparing it with HistoQC and concluded that our model works better in terms of predicting the artifacts. We also improved the HistoROI classification model by using the Supervised Contrastive Learning to generate better feature embeddings of the patches. We also noticed few shortcomings with the data preparation process in detail and provided possible solutions to overcome them.

5.1.2 Cell detection using Cell-Tissue Interaction

1. In conclusion, our findings in Section 3.6 demonstrate the effectiveness of incorporating the tissue segmentation model alongside a baseline algorithm to provide a broader context for cell classification. Notably, we observed significant improvements in the results when employing the cell segmentation method compared to the YOLOv8 approach for cell detection.

2. The best-performing model was one that leveraged the cell segmentation method in conjunction with the tissue segmentation predictions. This model was submitted during the validation phase of the challenge. We achieved a promising mF1 score of 0.67 on the validation dataset, which, at present, remains unpublished by the challenge organizers.

3. At the time of writing, our model has secured the 4th position on the leaderboard. These results underscore the efficacy of our proposed methodology, highlighting the importance of leveraging both cellular and tissue context in accurately detecting and classifying cells. Our performance in the challenge reflects the potential impact

of our research in advancing computational pathology and lays the foundation for further investigations in this field.

5.1.3 Quality Control tool for WSIs

Our study has presented a comprehensive framework for quality control in whole slide images (WSIs) by leveraging a combination of segmentation models. By incorporating blur level detection, tissue fold detection, tissue detection, and pen marker detection, we successfully generated a final usable mask for WSIs. The comparison of our results with the widely used HistoQC tool revealed a remarkable level of agreement, demonstrating the effectiveness of our WSI Profiler model as a quality control tool.

Through the integration of multiple segmentation models, our framework offers a robust approach for assessing the quality of WSIs. By accurately detecting blur levels, identifying tissue folds, distinguishing tissue regions, and detecting pen markers, our model addresses critical aspects of quality control in digital pathology and microscopy. The high level of agreement between our WSI Profiler model and HistoQC emphasizes its reliability and potential for improving quality control processes in the field of pathology.

The comprehensive framework presented in this study provides a valuable tool for researchers, pathologists, and clinicians working with WSIs. By automating the quality control process and providing a reliable assessment of image quality, our model streamlines the analysis workflow and enhances the accuracy and reliability of diagnostic evaluations based on WSIs.

5.2 Future Works

5.2.1 HistoROI: Histopathology specific preprocessing

As we discussed the shortcomings of the current data preparation pipeline in section 1.4.1, in the stage-2 of the project, we intend to implement the possible solutions discussed in section 1.4.1 and move towards developing a more robust Quality Control solution with additional task of generating segmentation masks.

5.2.2 Cell detection using Cell-Tissue Interaction

1. Joint Training: Investigating the joint training of the tissue and cell segmentation models is a potential avenue for improvement. By assigning appropriate weights to the corresponding loss functions, we can explore the synergy between these two tasks. It would be beneficial to experiment with various architectures beyond the DeepLabV3+ model we employed, in order to determine the most effective combination and validate the results.

2. Loss Function Exploration: Another area for exploration lies in the realm of loss functions. While our current approach employs the dice loss for training the cell segmentation model and cross-entropy loss for the tissue segmentation model, there exist a plethora of alternative loss functions that could potentially yield better performance. By systematically exploring and comparing different loss functions, along with thorough hyper-parameter tuning, we can identify the most suitable choice for each model and optimize their performance.

5.2.3 Quality Control tool for WSIs

1. Future research should focus on further investigating advanced architectures and techniques beyond the UNet++ model. This exploration has the potential to enhance the accuracy and efficiency of segmentation models utilized in the WSI Profiler.

2. To gain deeper insights and identify the strengths and weaknesses of the quality control tool developed, it is crucial to involve clinical pathologists in the evaluation process. Conducting a comparative evaluation between the disagreement observed with HistoQC and the disagreement detected by our quality control tool can provide valuable insights. Involving clinical pathologists will help assess the performance of the tool in relation to human expert judgment and identify areas for improvement.

3. In order to assess the practical impact of the findings and validate the developed quality control method, deployment in real clinical settings is necessary. The implementation of these methods in clinical settings will allow for their evaluation, validation, and assessment of usefulness in improving pathology workflows.

Bibliography

[1] Birgid Schömig-Markiefka, Alexey Pryalukhin, Wolfgang Hulla, Andrey Bychkov, Junya Fukuoka, Anant Madabhushi, Viktor Achter, Lech Nieroda, Reinhard Büttner, Alexander Quaas, et al., "Quality control stress test for deep learning-based diagnostic model in digital pathology," *Modern Pathology*, vol. 34, no. 12, pp. 2098–2108, 2021.

[2] Alexander I Wright, Catriona M Dunn, Michael Hale, Gordon GA Hutchins, and Darren E Treanor, "The effect of quality control on accuracy of digital pathology image analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 2, pp. 307–314, 2020.

[3] Maryam Haghighat, Lisa Browning, Korsuk Sirinukunwattana, Stefano Malacrino, Nasullah Khalid Alham, Richard Colling, Ying Cui, Emad Rakha, Freddie Hamdy, Clare Verrill, and Jens Rittscher, "Pathprofiler: Automated quality assessment of retrospective histopathology whole-slide image cohorts by artificial intelligence, a case study for prostate cancer research," *medRxiv*, 2021.

[4] Jeongun Ryu, Aaron Valero Puche, JaeWoong Shin, Seonwook Park, Biagio Brattoli, Jinhee Lee, Wonkyung Jung, Soo Ick Cho, Kyunghyun Paeng, Chan-Young Ock, Donggeun Yoo, and Sérgio Pereira, "Ocelot: Overlapped cell on tissue dataset for histopathology," 2023.

[5] "Ocelot 2023: Cell detection from cell-tissue interaction," 2023, Accessed on June 12, 2023.

[6] Yuri Tolkach, Tilmann Dohmgörgen, Marieta Toma, and Glen Kristiansen, "High-accuracy prostate cancer pathology using deep learning," *Nature Machine Intelligence*, vol. 2, no. 7, pp. 411–418, 2020.

[7] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[8] Alexander I Wright, Derek Magee, Philip Quirke, and Darren Treanor, "Incorporating local and global context for better automated analysis of colorectal cancer on digital pathology slides," *Procedia Computer Science*, vol. 90, pp. 125–131, 2016.

[9] Leo Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[10] Andrew Janowczyk, Ren Zuo, Hannah Gilmore, Michael Feldman, and Anant Madabhushi, "Histoqc: an open-source quality control tool for digital pathology slides," *JCO clinical cancer informatics*, vol. 3, pp. 1–7, 2019.

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

[12] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, "The elements of statistical learning. springer series in statistics," *New York, NY, USA*, 2001.

[13] Nadia Brancati, Giuseppe De Pietro, Daniel Riccio, and Maria Frucci, "Gigapixel histopathological image analysis using attention-based neural networks," *IEEE Access*, vol. 9, pp. 87552–87562, 2021.

[14] Mingxing Tan and Quoc Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

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

[16] John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, and Joshua M Stuart, "The cancer genome atlas pan-cancer analysis project," *Nature genetics*, vol. 45, no. 10, pp. 1113–1120, 2013.

[17] Qiong Liu and Ying Wu, "Supervised learning," 01 2012.

[18] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan, "Supervised contrastive learning," 2021.

[19] Carolyn Hutter and Jean Claude Zenklusen, "The cancer genome atlas: Creating lasting value beyond its data," *Cell*, vol. 173, no. 2, pp. 283–285, 2018.

[20] Glenn Jocher, Ayush Chaurasia, and Jing Qiu, "Yolo by ultralytics," 2023.

[21] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," 2018.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," 2015.

[23] Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li, "Dice loss for data-imbalanced nlp tasks," 2020.

[24] Ilya Loshchilov and Frank Hutter, "Sgdr: Stochastic gradient descent with warm restarts," 2017.

[25] Mahdi S Hosseini, Yueyang Zhang, and Konstantinos N Plataniotis, "Encoding visual sensitivity by maxpol convolution filters for image sharpness assessment," *IEEE Transactions on Image Processing*, vol. 28, no. 9, pp. 4510–4525, 2019.

[26] Adyn Miles and N Konstantinos, "Focuslitenn: High efficiency focus quality assessment for digital pathology," *arXiv preprint arXiv:2007.06565*, 2020.