Multiclass Classification and Semantic Segmentation of Colorectal Cancer Cells from Histopathology Images

Mahdi Islam , Musarrat Tabassum , Sirada Kittipaisarnkul , Sarita Mourya , Tuhinangshu Gangopadhyay

Università degli studi di Cassino e del Lazio Meridionale, Italy

Abstract

Colorectal Cancer is known to be one of the leading reasons for deaths, with the death toll rising every year. Thus early detection of this cancer is essential for starting treatment at a premature stage, so as to avoid any serious consequences in the latter stage. In this report, we have proposed three different techniques to exploit histopathology images for early detection of Colorectal Cancer in patients. The first technique involves creating multiple pipelines for segmenting the images using advanced image processing algorithms, like morphology, unwanted object removal, contour detection, watershed segmentation, etc. The average dice score is 82%. In the second technique, we have employed various methods like Local Binary Patterns (LBP) and Gabor filters for feature extraction from the images and then using Machine Learning algorithms for classifying the images into different classes, like Normal, Polyp, Low-grade IN, High-grade IN, Serrated Adenoma and Adenocarcinoma. The classification accuracy for this solution is 83%. The final technique involves using Deep Learning models with different backbones for performing segmentation of the histopathology images. After further experimentations, it can be concluded that the best performance for histopathology image segmentation is given by the UNet model with an EfficientNet-B2 backbone (with a dice score performance of 93%), owing to its efficiency and capability in learning the deep semantic representative features.

Keywords: Colorectal Cancer, Histopathology Image Classification and Segmentation, Advanced Image Analysis, Machine Learning and Deep Learning

1. Introduction

Colorectal cancer (CRC), also known as bowel cancer, is the second highest cause for death related to cancer globally. According to the World Health Organization, in 2020, 1.9 million new cases and 930,000 deaths were estimated globally because of CRC. The results of prediction of CRC by 2040 shows that new cases will increase by 63% and deaths will increase by 73%. Colorectal cancer is a type of cancer that develops in the rectum or the colon of the digestive tract or system. It starts as a noncancerous clump of small muscle of cell type called polyps, and if left untreated, it sometimes turns into cancer. Bowel diseases such as ulcerative colitis and chronic digestive diseases increase the chance of colon cancer. CRC typically originates from precancerous growths called adenomatous polyps. Colorectal Polyps refers to unwanted growth on the mucosal surface. It has a similar shape to normal cells. Moreover, there are various polyp types. Intraepithelial neoplasia (IN) is the most critical precancerous stage. It has different luminal sizes and shapes, dense arrangement, the nuclei are enlarged and vary in size. Adenocarcinoma is a malignant digestive tract tumor with irregular distribution of luminal structure. The nuclei are large in this stage. Serrated adenoma is a very uncommon lesion, the shape is thought to be similar to the colon with tubular or cerebral openings. Early-stage CRC often presents without symptoms. However, the risk of developing the disease increases significantly with age, with a higher prevalence observed in individuals above 50. Histopathological examination remains the gold standard for detecting this disease. As in the present generation the unhealthy lifestyle is widespread as the fashion of relying on processed food, heavy alcohol consumption, smoking, and thus, colon cancer is undeniably a rampant concern in today's society. It's among the most common types of cancer globally, with incidence rates varying across different regions and populations. Detection of colon cancer in

the early stages improves treatment results and increases the probabilities of successful recovery. Recognizing issues or health conditions in their early stages every so often offers several advantages. That is, catching it early allows for more effective interventions, possibly preventing it from intensifying into a bigger problem.

Treatment with Hematoxylin and Eosin (H&E) is a common approach which stains tissue to show the inclusion between nucleus and cytoplasm, showing location of lesion. However, there are problems in the diagnostic process: The diagnostic result becomes subjective to each medical expert and human error. Nowadays, Computer-Aided Diagnostics (CAD) helps to improve accuracy in the aspect of image segmentation. Integrating image processing techniques, machine learning (ML) and deep learning (DL) methods with histopathological data can definitely augment the effectiveness and precision of early diagnosis and classification of colorectal cancer. ML algorithms can inevitably extract appropriate features from histopathological images, such as the texture, shape, and spatial arrangement of cells and tissues. DL models learn classified representations straight from raw image data, supporting them to identify complex patterns that may be difficult for traditional algorithms to distinguish. Histopathological images provide high-resolution information of tissue morphology, cellular structures, and pathological features, which is significantly supportive for the ML algorithms to detect elusive changes and revealing of the cancerous lesion with better accuracy. It is helpful in analyzing the histopathological features such as tumor size, shape and density aiding in early intervention and enhanced patient results and aids in the treatment planning which boost the chance of survival.

Patients with Colorectal cancer have personalized treatment according to the stage of tumor determined by biomarkers, clinical data, histopathological analysis, molecular pathology of tumor cells. Treatment with Hematoxylin and Eosin (H&E) is a common approach which stains tissue to show the inclusion between nucleus and cytoplasm, showing location of lesion. Hematoxylin stains cell nuclei in purple-blue hue, while Eosin stains cytoplasm and extracellular in pink-red hue. The advancement in Machine learning and Deep learning in Computer-Aided Diagnostic (CAD) helps to improve accuracy in the aspect of image segmentation. However, medical experts need cross validation to confirm the diagnostic and tumor staging, which can lead to human error since it is subjective to each medical experts

In this report, we propose 3 solutions for classifying and segmenting 6 tumor types of Colorectal cancer stages which are Normal, Polyp, Low-Grade and High-Grade Intraepithelial Neoplasia (IN), Adenocarcinoma, Serrated Adenoma. In the first solution, we developed an advanced Image Processing based pipeline(s) to perform histopathology image segmentation. The second solution uses feature extraction techniques, followed by Machine Learning based methods for classifying the histopathology images into one of the six classes. Finally, the last pipeline employs Deep Learning for performing segmentation of the histopathology images.

2. Related Works

Diagnosing colorectal cancer through histopathological images is a complex task that has garnered significant research interest. Various studies have focused on developing image processing, machine learning and deep learning methods specifically for segmenting and classifying the lumens in these images. In this section, we present the literature review that highlights key studies relevant to this topic.

In 1985, Fenoglio-Preiser and Hutter [1] discussed the pathologic diagnosis and clinical significance of colorectal polyps, shedding light on the importance of accurate diagnosis and management of these lesions in preventing colorectal cancer. Their work underscored the critical role of histopathological examination in identifying and characterizing colorectal polyps, which can serve as precursors to malignant tumors. However, challenges remain in accurately distinguishing between benign and malignant polyps, highlighting the need for improved diagnostic strategies and technologies to enhance patient care and outcomes.

Gurcan et al. in 2009 [2] provided a comprehensive review of histopathological image analysis, covering various methodologies, including feature extraction, pattern recognition, and classification algorithms. Their review

underscored the significance of computational approaches in interpreting complex tissue structures and identifying cancerous regions, contributing to advancements in cancer diagnosis and prognosis. In 2011, Pietikainen et al. [3] discussed the application of computer vision techniques, specifically local binary patterns (LBP), in analyzing histopathological images. LBP offers a robust method for texture analysis, facilitating the identification of abnormal cell patterns indicative of cancerous tissues. This approach holds potential for improving the accuracy of cancer diagnosis through automated image analysis. However, one drawback of LBP is its sensitivity to variations in image acquisition parameters such as resolution and illumination. Inconsistencies in these parameters can affect the performance of LBP-based analysis, leading to potential misinterpretation of histopathological features and compromising the reliability of the diagnostic outcomes.

Rathore and Iftikhar in 2016 [4] presented CBISC, which leverages epithelial cell morphology for segmenting and classifying colon biopsy images. The segmentation module detects elliptic cells and calculates unique features for each pixel, with optimization through a genetic algorithm. Classification relies on gray-level features, with support vector machines achieving reasonable results. This approach underscores the importance of morphology-based segmentation for accurate diagnosis. Mármol et al. in 2017 [5] presented a comprehensive overview of colorectal carcinoma, encompassing its epidemiology, pathogenesis, clinical manifestations, diagnostic methods, and therapeutic interventions. The review offered valuable insights into the multifactorial nature of colorectal cancer, emphasizing the importance of early detection and personalized treatment strategies. Later in the same year, Chaddad and Tanougast [6] explored texture analysis of abnormal cell images for predicting the continuum of colorectal cancer by leveraging advanced image processing techniques. This method enabled early detection and characterization of colorectal cancer, facilitating more targeted and personalized treatment strategies.

Cao et al. [7] proposed a novel approach to enhance the performance of transfer learning without fine-tuning for breast cancer histology images. By leveraging dissimilarity-based multi-view learning, their method achieved notable improvements in classification accuracy, providing a promising avenue for more efficient and effective cancer detection strategies. Rathore et al. [8] proposed a comprehensive multi-step gland segmentation method, leveraging ellipsoidal modeling of tissue components. Their aim is to improve cancer detection and grading by capturing cellular morphology, spatial glandular patterns, and texture through the extraction of multi-scale features. These features are classified into gland-based, local-patch-based, and image-based categories and utilized in a hierarchical ensemble-classification approach.

Kurmi et al. in 2019 [9] investigated tumor malignancy detection using histopathology imaging techniques. This method combines handcrafted features and shape features using a bag of visual words (BoW) for image classification. It involves a multistage segmentation technique to localize nuclei in histopathology images, starting with stain decomposition and histogram equalization to enhance the nucleus region. Key point extraction is performed using the fast radial symmetry transform, followed by nuclei region estimation with normalized graph cut, and boundary estimation via a modified gradient approach. Features from these localized regions are extracted and categorized into handcrafted and shape features using BoW. Tested on the Bisque and BreakHis datasets, the method achieves average accuracies of 93.87% and 96.96%, respectively, suggesting enhanced diagnostic performance by effectively integrating both feature types for image classification in biomedical imaging, particularly for cancer diagnosis and grading.

Gupta et al. (2021) [10] proposed a methodology for breast cancer detection from histopathology images using modified residual neural networks. By optimizing deep learning architectures, their approach achieved impressive accuracy in detecting breast cancer, highlighting the potential of deep learning techniques in improving cancer diagnosis outcomes. In a similar vein, Babu et al. [11] introduced an optimized method addressing the challenges posed by variability in image characteristics and magnifications. He proposed a magnification-independent segmentation approach. It combines connected component area and double density dual tree DWT coefficients for segmentation, followed by fuzzy c-means clustering for feature reduction. An artificial neural network optimized with salp swarm optimization classifies images into normal and abnormal. Evaluation across four datasets with varied magnifications show significant improvements over existing techniques, promising reliable cancer detection.

Ben Hamida et al. [12] addressed the challenges of histopathological image segmentation with weakly supervised learning using attention gates. Their proposed model achieved improved image segmentation results, enhancing the accuracy of colon cancer detection. It addresses the challenges in using deep learning models for histopathological image segmentation tasks. The authors introduce enhanced models of the Att-UNet, proposing various configurations for skip connections and spatial attention gates within the network. These gates facilitate the training process by helping the model avoid learning irrelevant features. The Alter-AttUNet model, which includes these modules, achieves increased robustness and improved image segmentation results. Talukder et al. [13] proposed a machine learning-based approach involving a hybrid ensemble feature extraction model designed to efficiently identify lung and colon cancer. This model integrates deep feature extraction with ensemble learning and employs high-performance filtering specifically tailored for cancer image datasets. The model's performance was evaluated using the LC25000 histopathological dataset, which includes lung and colon cancer images. The results indicate that the hybrid model achieved remarkable accuracy rates: 99.05% for lung cancer, 100% for colon cancer, and 99.30% for both lung and colon cancers. These findings demonstrate that the proposed strategy significantly outperforms existing models, highlighting its potential application in clinical settings.

Tharwat et al. [14] conducted a comprehensive study on colon cancer diagnosis utilizing machine learning and deep learning techniques. They investigated various modalities and analysis techniques, such as support vector machines (SVMs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and other deep learning architectures. These approaches involved data preprocessing, feature extraction, and model training using machine learning algorithms or deep neural networks. The study provided insights into the potential applications of these methods in improving diagnostic accuracy. Sakr et al. [15] proposed an efficient deep learning approach specifically tailored for colon cancer detection. Their method leveraged advanced neural network architectures, such as convolutional neural networks (CNNs) or recurrent neural networks (RNNs), trained on large-scale histopathological image datasets. By employing deep learning techniques, their approach demonstrated promising results in accurately identifying colon cancer tissues from histopathological images.

Furthermore, Shi et al. in 2023 [16] introduced the EBHI-Seg dataset, a valuable resource for image segmentation tasks in enteroscopy biopsy histopathological hematoxylin and eosin images. This dataset is made publicly available, which contributed to the advancement of segmentation techniques in the field of histopathology.

In conclusion, recent advancements in histopathological image analysis have significantly enhanced our ability to detect and diagnose cancer. By leveraging innovative computational techniques, such as transfer learning, texture analysis, and deep learning, researchers have made substantial progress in automating cancer detection processes and improving diagnostic accuracy. Continued research in this field can further advance our understanding of cancer pathology and help improve patient outcomes through early detection and treatment strategies.

3. Dataset Description

We have used the publicly available EBHI-Seg dataset [16] with 4,456 images, consisting of 2,228 histopathology section images and 2,228 ground truth images. The input images (and their corresponding ground truth segmentation) have a size of 224×224 pixels in Portable Network Graphic (PNG) format. The dataset is available at: https://doi.org/10.6084/m9.figshare.21540159.v1

This dataset consisted of 5 types of Intestinal biopsy; Normal, Polyp, Intraepithelial Neoplasia (IN), Adenocarcinoma, Serrated Adenoma, as described below:

1. **Normal:** This category contains non-diseased images of the colorectal tissue sections; that is, they do not have any kind of infection, when viewed under a microscope. They have a very regular lumen structure. This class has 76 histopathological images in the dataset.
2. **Polyp:** Polyps are unwanted growth of a mass/blob on the mucosal surface of the body. These polyps have a structure similar to that of the normal images, but their histopathological structures are different. The polyps are not

necessarily malignant. Infact, in most cases they are benign, however, their detection is important, as they might develop into a malignant cancer over the years. There are a total of 474 images of this class in the dataset.

3. **Intraepithelial Neoplasia (IN):** This is the most critical type of precancerous lesion. Its histopathological structure shows heavily branched adenoid structure, dense arrangement and highly irregular luminal shapes and sizes, because the nucleus has been highly enlarged. These images are further classified into two types: Low-grade INs and High-grade INs; with the latter displaying a more pronounced structural misalignment than the former one. Low-grade IN class has 637 images, and High-grade IN class has 186 images in the dataset.

4. **Adenocarcinoma:** Adenocarcinoma is a very malignant type of digestive tract tumors that can develop from a polyp, and they have a very irregular distribution of the lumen. It is very difficult to delineate the structures of the lumen during observations. The image count of this class in the dataset is 795 images.

5. **Serrated Adenoma:** They are a very uncommon type of lesions. The surface appearance of this type of lesions are not very well defined, but are considered similar to that of the colonic adenomas with a tubular or cerebral crypt opening. This class has only 58 images in the entire dataset.

Sample images of each of these categories are displayed below in Fig. 1. In histopathology images, the target is to segment the lumen. The lumen borders are formed by cells.

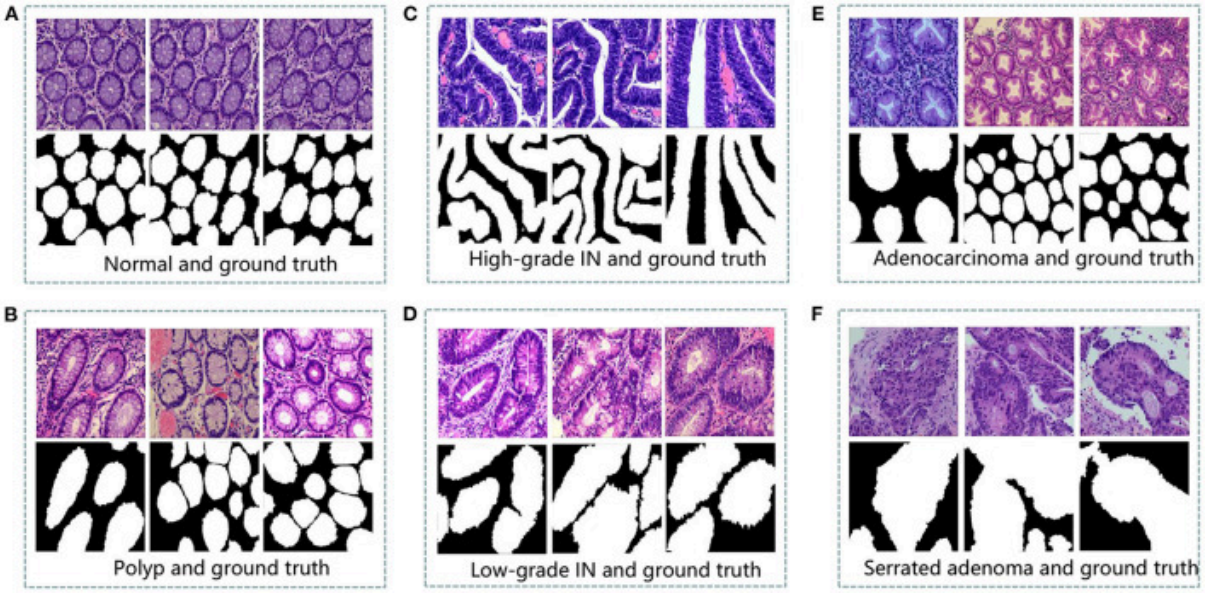


Fig. 1: Sample images of each category of histopathological images, with their corresponding segmentation

4. Methodology

In this section, we have discussed the detailed description of the methodology we adapted for performing our task, including the evaluation results of these methodologies. Particularly, this section has three subsections:

1. Image Processing
2. Machine learning
3. Deep Learning

In image processing, we deal with segmenting the histopathology images using only image processing and image analysis techniques. In Machine Learning, our target is to classify the images into their corresponding types. And in Deep Learning, our aim is to use Deep Learning based models for segmenting the histopathology images. For evaluation purposes, we propose to use the following metrics (TP: True Positive, FP: False Positive, FN: False Negative and TN: True Negative):

$$Precision\ (P) = \frac{TP}{TP+FP}$$

$$Recall\ (R) = \frac{TP}{TP+FN}$$

$$Jaccard\ Similarity\ (J) = \frac{TP}{FP+TP+FN}$$

$$Dice\ Score\ (D) = \frac{2TP}{FP+2TP+FN}$$

$$Accuracy\ (A) = \frac{TP}{TP+FP+TN+FP}$$

For the segmentation tasks, we used Precision, Recall, Jaccard Similarity, Dice Score and Accuracy; and for the classification task, we used Precision, Recall and Accuracy.

4.1. Image Processing

In this section, we proposed to solve the problem of Histopathology image segmentation by proposing a pipeline that is based only on Image processing and Image Analysis concepts. More precisely, we propose a set of pipelines, with each pipeline focusing one class of histopathology images; that is, we require prior knowledge about the classes of the images for segmenting them. However, these pipelines are not very different from each other. Fig. 2 shows the general pipeline that has been used, and this pipeline has been tweaked for each class, depending on the image complexity. In this flow diagram, the blue arrows represent the operations that are performed by all the pipelines, the black arrow represents the operations performed by some pipelines and skipped by others, the red lines show the operations performed in place of those skipped pipelines, and the pink arrows represent branching and merging operations. These pipelines make an assumption that a lumen will be segmented, only if a major portion of the lumen is present in the image. The reason for this is explained later.

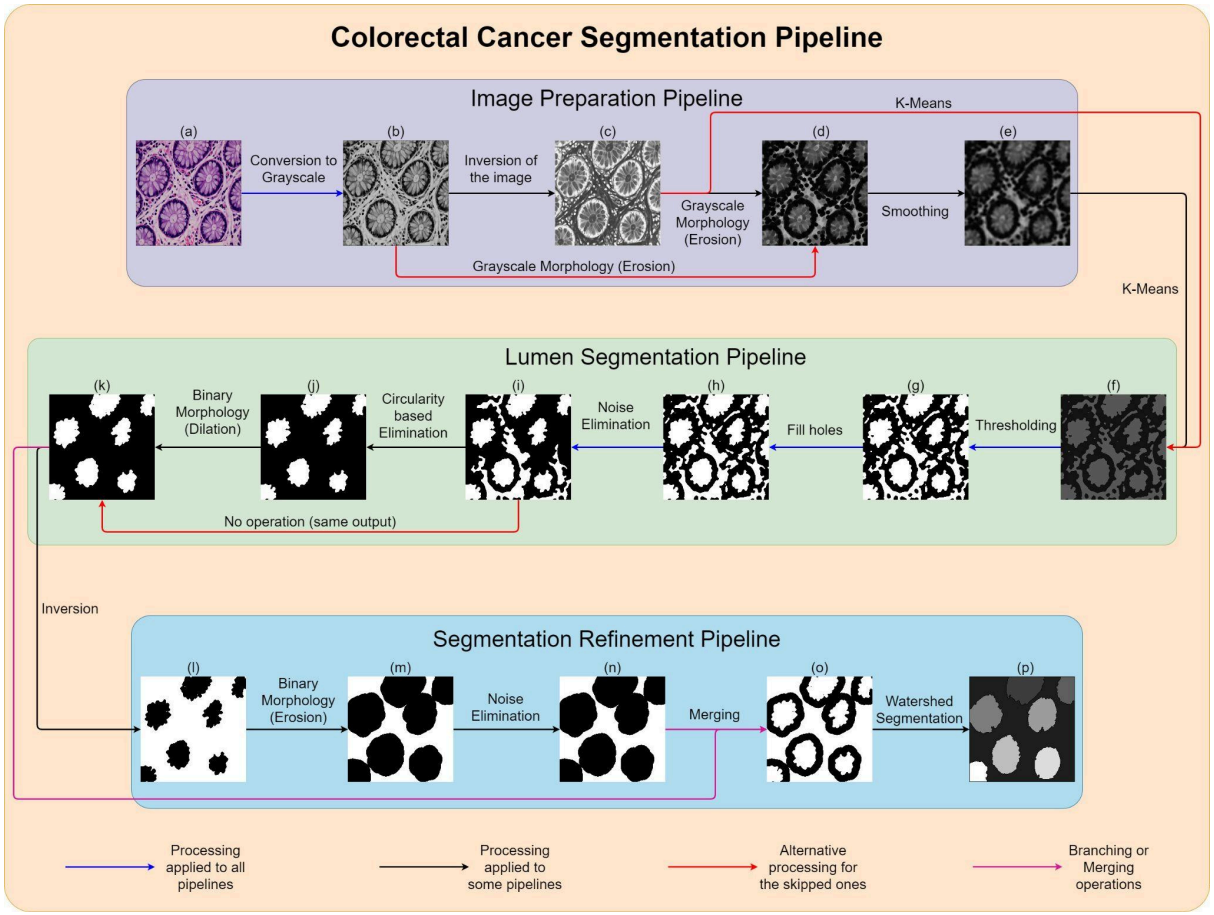


Fig. 2: The general pipeline describing the steps followed for performing the segmentation. It is divided into three sub-pipelines: Image Preparation Pipeline, Lumen Segmentation Pipeline, and Segmentation Refinement Pipeline.

The complexity of this segmentation task is evident from Fig. 1, which clearly shows the high variation of these types of images. The complicacies of the images are due to the:

1. Variation in shape and size of the lumen
2. Variation in the count of lumen in each image and the distance between them
3. Lack of a proper or strong boundary around the lumen
4. Lumens are completely surrounded by cells
5. Variation in color, intensity and contrast among the images

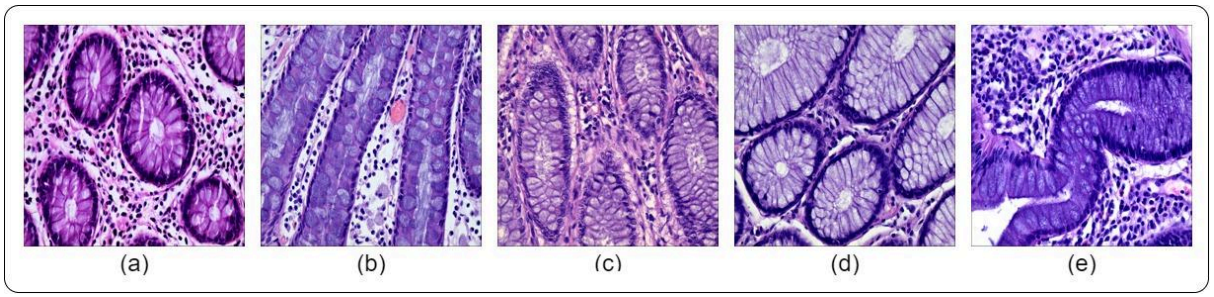


Fig. 3: Variations in the lumen structures of Histopathological images in the class 'Normal'

Even the images in a single class do not adhere to a single variation. For example, Fig. 3 shows five sample images, all belonging to the class "Normal". The extent of variation of images in each class is well depicted by this image. We now discuss the pipeline in detail.

The entire pipeline is divided into three major sub-pipelines, namely the Image Preparation Pipeline, the Lumen Segmentation Pipeline and the Segmentation Refinement Pipeline. The Image Preparation pipeline involves applying essential operations to the input image to convert it into a form, essential for segmentation. This pipeline is present to deal with the variation in images (mainly tackle the irregular variation in color, intensity and contrast, the irregular distribution of cells around the lumen and the cracked boundaries), and effort has been made to convert it to a standard form for segmentation purposes.

4.1.1. Image Preparation Pipeline

I. Conversion to Gray-Scale: The color images were converted to Gray-Scale images because the RGB channels did not show significant characteristics that could have been useful towards the segmentation. Color space conversion to HSV or LAB was also not found useful.

II. Inversion of the Gray-Scale image: The inversion of grayscale image was proposed to deal with images where the surrounding of the lumen is brighter than the lumen center (for example, when the surrounding is sparse, having a less number of cells, or due to the difference in intensity). As can be expected, this operation is not applied on all images, but only on the images which satisfy the above mentioned criteria. Inversion will cause the lumen to be the brighter object, which is a requirement for our pipeline. Inversion is given by the following equation (Here $gray$ is the grayscale image to be inverted):

$$inverted\_image = 255 - gray$$

III. Grayscale Morphology (Erosion): After ensuring that the lumen is brighter than the surrounding, a strong erosion is applied on the images to enlarge the darker regions (which is generally the cells surrounding the lumen). This step ensures that most of the cells that form the cracked lumen boundary, enlarge and merge, thus strengthening the boundary. This erosion operation is given by (where A is the grayscale image and B is the structuring element, centered on x):

$$A \ominus B = \{x | B_x \subseteq A\}$$

IV. Smoothing: Then a smoothing filter is applied to the above output to smoothen the image and join the cells in the lumen boundary that are still separated after the above morphological step. A custom filter has been designed for this purpose:

$$\frac{1}{75}\begin{bmatrix} 3 & 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 & 3 \\ 3 & 3 & 1 & 3 & 3 \\ 3 & 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 & 3 \end{bmatrix}$$

The filter is designed to give more importance to the surrounding object pixels than the center pixel. This is particularly helpful when the center filter element is on the boundary crack and the neighboring elements are on cells; the result is the merged boundary. The 1/75 multiplier in the filter is applied for normalization purposes. This filter has been found to suit our case better than the predefined smoothing filters like Gaussian filter and Median filter.

After ensuring that the issues due to the intensity variations have been resolved, and the cracked border have been merged, we now move on to the Lumen Segmentation Pipeline. A small thing to be noted is that, after the strong grayscale erosion, the lumen centers in the new image are smaller than that in the original image. This will be taken into account in the later stages of the overall pipeline.

4.1.2. Lumen Segmentation Pipeline

V. K-Means: K-Means algorithm is then applied, with the number of clusters as 2. This will divide the image into two regions: the brighter and the darker region. K-Means was found to do a much better job in segmenting the image into two regions, than any thresholding method.

VI. Thresholding: The job of thresholding here is just to binarize the K-Means output. We used Triangular Thresholding, since now we just have two distinct peak points in the image histogram.

VII. Filling holes: This is an algorithm developed to fill the gaps inside any closed contour. This step is required to ensure that the latter stages do not affect the already segmented lumen centers. The algorithm for filling holes is depicted in Fig. 4.

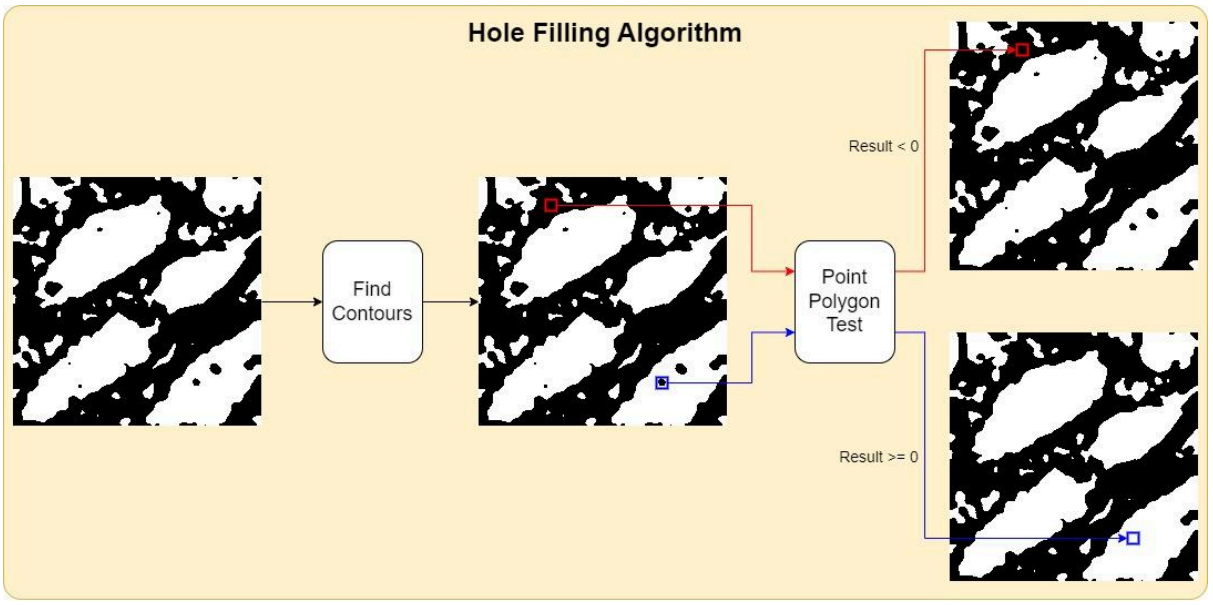


Fig. 4: The Hole Filling Algorithm used for filling the holes in the segmentation

The above algorithm finds the contours of the input thresholded image. It then performs the point polygon test on every pixel of the input, which returns a positive value in case the current point under consideration is inside a closed contour; otherwise it returns a negative value. If the value is positive then it is assigned to the foreground class.

VIII. Noise Elimination: This algorithm removes the small background elements, which can be considered as noise/unwanted elements. Fig. 5 represents the noise elimination algorithm used.

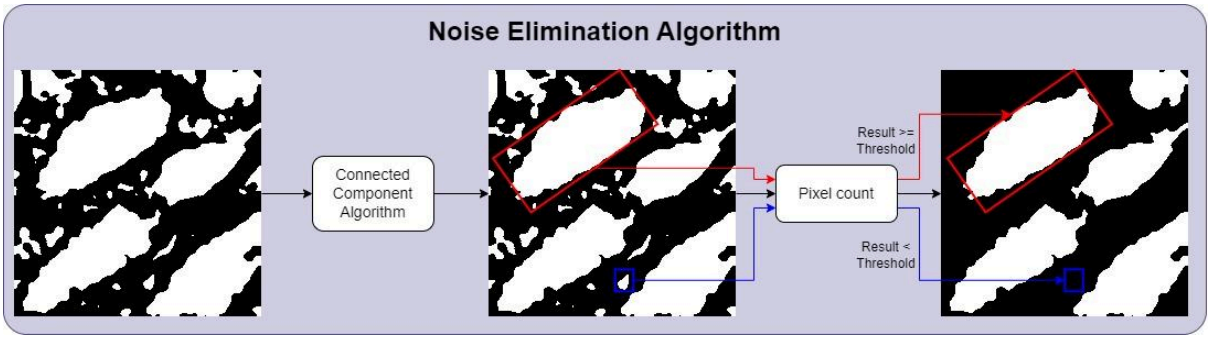


Fig. 5: Noise Elimination Algorithm for removing unwanted objects based on size

The algorithm uses connected components to get the individual objects in the segmentation mask. Then the pixel count of each object is used to compare with a threshold and remove objects based on this threshold. This threshold is itself a variable, which is different for each image and is given by the following formula:

$$threshold = \frac{sum(max\_cir\_list)}{5*scale}$$

Where, max_cir_list is the list of pixel counts of five largest objects in the mask, and $scale$ is a parameter whose value is decided experimentally. Thus this threshold leverages the average size of lumen centers to set the threshold. This is essential because the lumen centers are of different sizes in different images. As a result, one single threshold value can not satisfy all the images.

This Noise Elimination step is where the partially visible lumen centers (at the corner of the image) are removed from the image, after they are considered as noise, due to their small size (or small pixel count). As a result, our pipeline incorporates those lumens in the segmentation, which have a significant portion of the lumen center located within the image; and this explains our assumption stated previously.

IX. Circularity based elimination: This algorithm uses the circularity criteria to eliminate the unwanted elements, which could not be removed with the above size-based noise elimination algorithm. Fig. 6 shows the circularity based algorithm employed.

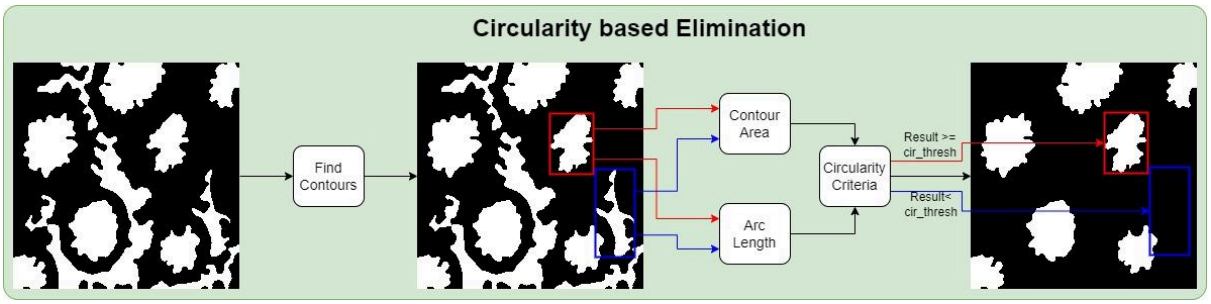


Fig. 6: Circularity based elimination for removing objects based on circularity

This algorithm finds the contours of the objects in the mask, followed by leveraging the contour area and arc length values to calculate the circularity with the below mentioned formula (where $A = Contour\ Area$ and $P = Arc\ Length$) and then removing the objects which are below a specified circularity threshold value.

$$Circularity = \frac{4\pi A}{P^2}$$

X. Binary Morphology (Dilation): Then binary morphology is applied to enlarge the remaining contours, which are the expected lumen centers. This is to compensate for the reduction in size of the lumen center, due to the

grayscale erosion in the previous pipeline. The dilation operation is given by (where A is the grayscale image and B is the structuring element, centered on x):

$$A \oplus B = \{x | B_x \cap A \neq \phi\}$$

After the Lumen Segmentation Pipeline, the output should have bright pixels corresponding to the lumen centers. However, the current segmentation is not completely accurate, due to the irregularity in lumen shape and size, and the complex operations included in the pipeline, which returned a shrinked version of the expected segmentation. The final dilation in the above pipeline is not enough for getting the expected lumen size due to the randomness in the lumen border width. Thus, we require the Segmentation Refinement pipeline to further convert the current output to a more accurate segmentation.

4.1.3. Segmentation Refinement Pipeline

XI. Inversion: The output of the previous pipeline is inverted; so the white pixels now contain two elements: the background pixels and the lumen border pixels.

XII. Binary Morphology (Erosion): A strong erosion is applied to the inverted image so as to exclude the lumen border pixels from the white pixels. The white pixels now contain pure background elements.

XIII. Noise Elimination: Some noise or small artifacts may also be introduced into the inverted image due to the erosion operation. They are removed using the same noise elimination algorithm, as described previously.

XIV. Merging: Now the output of the previous pipeline is merged with the above noise eliminated inversion. The output is the white pixels containing the lumen center and the background regions. The black region contains only the lumen boundary.

XV. Watershed Segmentation: Finally, the region-growing based Watershed Segmentation has been applied to the merged output to get the desired output. This algorithm will expand the lumen center pixels and the white background pixels (the eroded inversion) until they converge and stop at the lumen border.

The output of the watershed segmentation marks the end of our general segmentation pipeline. Now, we will discuss the class wise pipelines (Fig. 7), which is a modified version of the general pipeline.

4.1.4. ‘Normal’ and ‘Polyp’ classes

For both ‘Normal’ and ‘Polyp’ classes, we have proposed two different pipelines. One pipeline deals with the grey images, and the other pipeline focuses on the inverted grey images. Two different pipelines are essential to deal with the variations in intensity and cell distribution in the background of the histopathology image. This is an important necessity for our pipeline to execute properly. The segmentation results from the two pipelines are then given to a circularity based selection algorithm and its output is considered as the final segmentation output. The Circularity based Selection algorithm is described in Algorithm 1. Fig. 7 (a) shows the entire segmentation process for the ‘Normal’ and the ‘Polyp’ class. Fig. 8 shows samples of the segmentation quality returned by our pipeline.

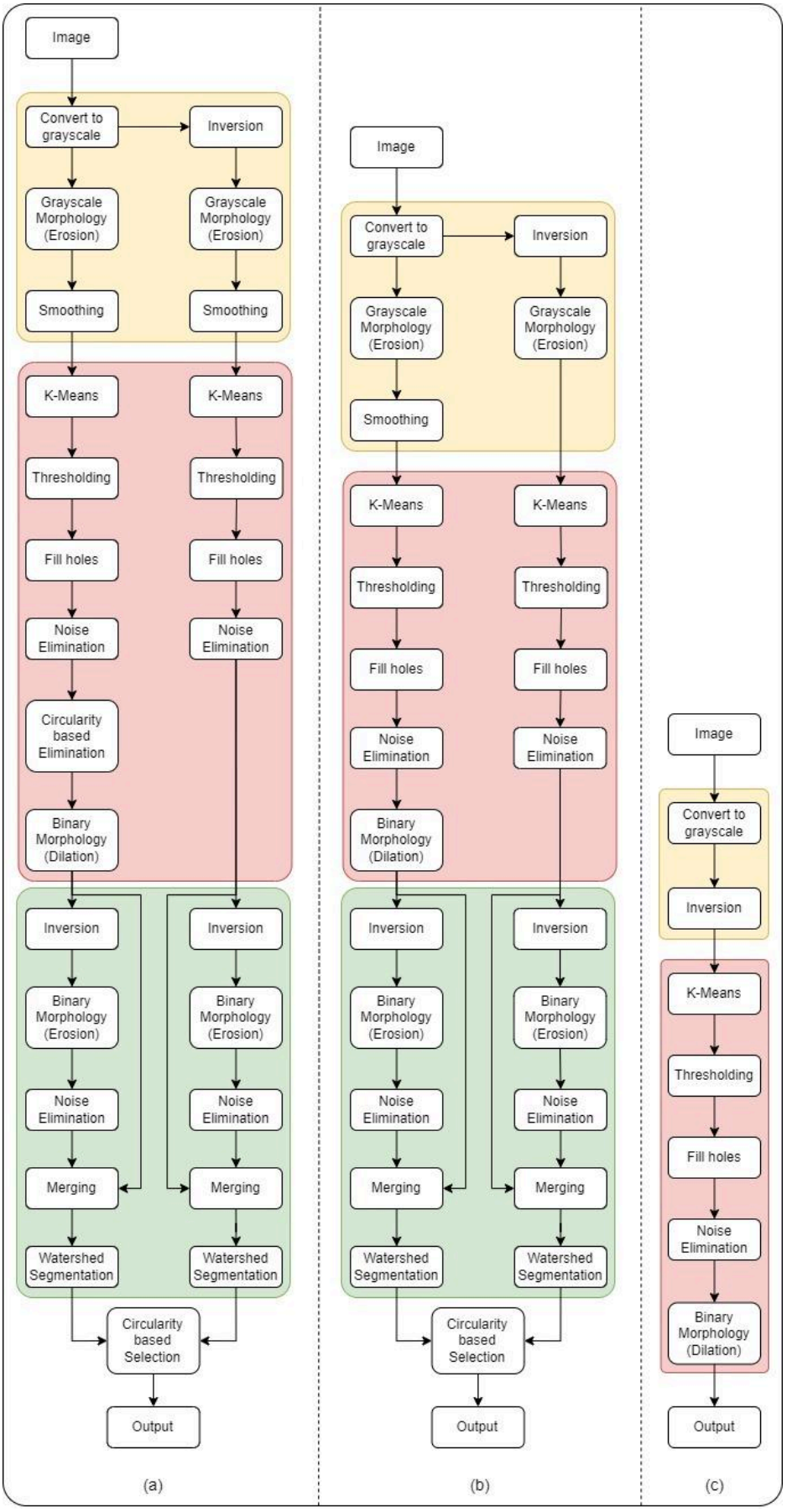


Fig. 7: Class-wise pipeline for each class. (a) for classes 'Normal' and 'Polyp', (b) for classes 'Low-grade IN' and 'Serrated Adenoma', and (c) for classes 'High-grade IN' and 'Adenocarcinoma'. All the pipelines are derived from the previously described general pipeline in Fig. 2. The yellow box represents the Image

Preparation Pipeline, the red box represents the Lumen Segmentation Pipeline and the green box represents the Segmentation Refinement Pipeline**.**

ALGORITHM 1: Circularity based Selection

Input:
$seg\_pipeline1$: Output of the first segmentation pipeline
$seg\_pipeline2$: Output of the second segmentation pipeline
cir_list1: Circularity values of all segmented objects in $seg\_pipeline1$
cir_list2: Circularity values of all segmented objects in $seg\_pipeline2$

Output:
$final\_seg$: The selected output among the two segmentation outputs

1. $cir1 = \frac{sum(cir\_list1)}{len(cir\_list1}$

2. $cir2 = \frac{sum(cir\_list2)}{len(cir\_list2)}$

3. $if\ cir1 > cir2$:
4. $\quad final\_seg = seg\_pipeline1$

5. $else$:
6. $\quad final\_seg = seg\_pipeline2$

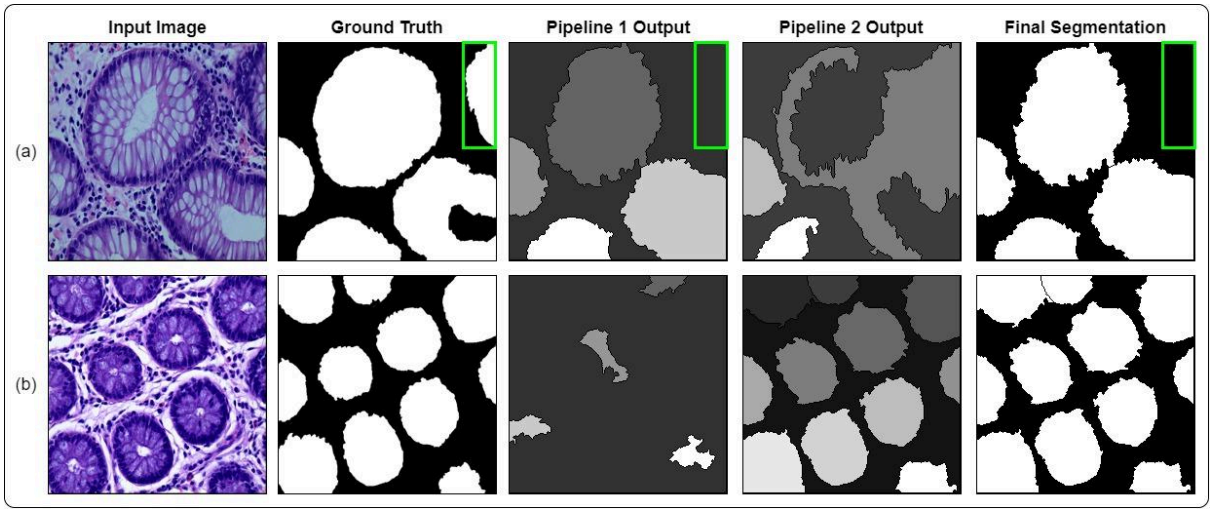


Fig. 8: Segmentation output by our proposed pipeline for two input images of 'Normal' and 'Polyp' class type

Fig. 8 demonstrates that both the pipelines are necessary for getting the final segmentation, as well as shows the importance of the Circularity based Selection algorithm. Fig 8(a) shows an input image with a dense cell distribution pattern outside the lumen and thus the first pipeline is suitable for its segmentation, whereas for Fig. 8(b), the second pipeline is required (that is, the inversion of the greyscale image is necessary before further processing) because the background intensity is higher than the lumen intensity. Furthermore, our segmentation output for pipeline 1 does not include the lumen at the top right corner in Fig 8(a) (the green box region). This is due to the assumption stated previously, about the size of the lumen center being considerably large in the input image for it to be included in the segmentation output.

An alternative approach to use a single pipeline (instead of using two pipelines), and handle the issue of variations in intensity and cell distribution in the background, would be to calculate the mean intensity difference between the lumen center region and the background region in the image, and proceed with the gray image or the inverted gray image based on a threshold value. However, this would require prior knowledge (or an assumption)

about the position and shape of the lumen centers for calculating the mean, which we lack. As a result, this approach could not be taken into account.

4.1.5. 'Low-grade IN' and 'Serrated Adenoma' classes

For 'Low-grade IN' and 'Serrated Adenoma' classes, we again proposed two different pipelines and combined their results using the Circularity based selection algorithm. As before, the first pipeline deals with grayscale images and the second pipeline deals with the inverted grayscale images, for reasons similar to the one mentioned above. The best output is again selected using the Circularity based Selection algorithm. Fig. 7(b) shows the segmentation pipelines for the classes 'Low-grade IN' and 'Serrated Adenoma' classes. As can be seen, the circularity based elimination criteria is no longer needed for segmenting these classes. Fig. 9 shows the segmentation returned by this combined pipeline. For input 9(a), pipeline 1 outputs an additional object (in the green box region). This is due to the resemblance of this region with the lumen pattern: a bright central region, surrounded by densely populated cells; and the elimination algorithms in our pipeline failed to remove this. For input 9(b), pipeline 2 outputs a better segmentation than pipeline 1. This is because of the large white region in the top left corner of the input image, which made the actual lumen region darker, leading to a poor performance by pipeline 1. The pipeline 2 segmentation does not consider the top left corner white region as a part of the lumen.

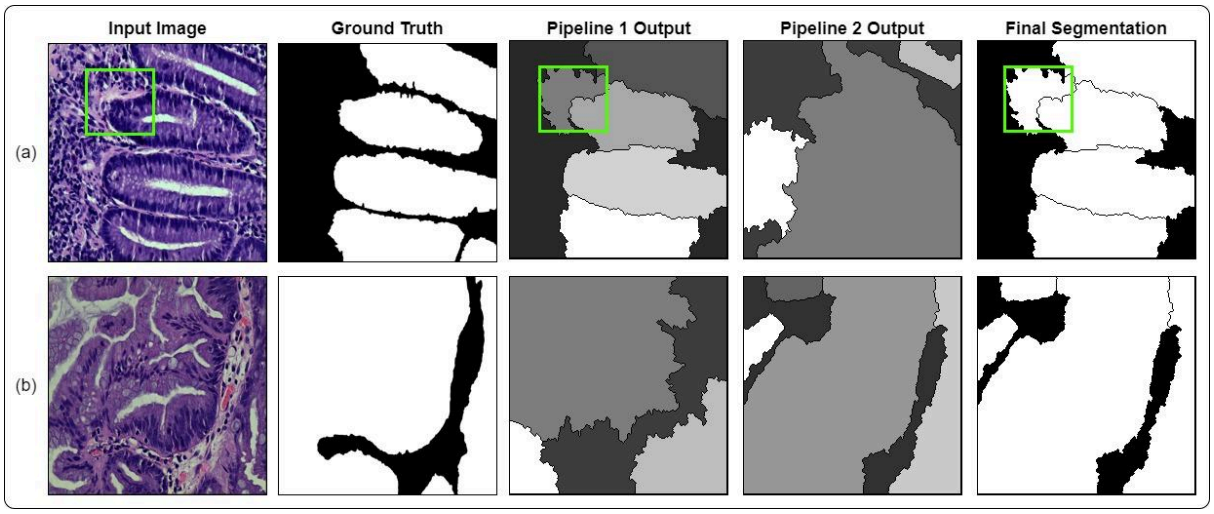


Fig. 9: Segmentation output for two input images of 'Low-grade IN' and 'Serrated Adenoma' class type

4.1.6. 'High-grade IN' and 'Adenocarcinoma' classes

Finally, for the last two classes 'High-grade IN' and 'Adenocarcinoma', only one pipeline was proposed and was found to be enough. This pipeline is much simpler than the previous pipelines and is presented in Fig. 7(c). This pipeline does not even employ the Segmentation Refinement Pipeline (which is described in Fig. 2). This is because this final sub-pipeline was proposed for segmenting refined lumen borders, and these two classes, being highly distorted in shape and size, don't have a proper/definite lumen border. Furthermore, after multiple experimentations, it was found that the Grayscale Erosion step and the following Smoothing steps are also not required for these two classes. This is again due to the lack of definite shape, size and borders (as mentioned earlier, these two operations were initially proposed for fixing the problem of cracked lumen borders in the images). Also, using these morphology and smoothing operations on a highly distorted image can be destructive, resulting in further loss of features. Fig. 10 shows the segmentation output returned by this pipeline. Input 10(a) clearly shows a lack of border around the lumen, thus leading to some minor overflows in the segmentation. Input 10(b) shows some presence of the lumen border, thus returning a better segmentation than the previous image. However, it skips the elongated white region inside the lumen. More precisely, our pipeline is designed to skip the white regions from the input images. But there is some ambiguity in the ground truth segmentation, for example, in image 10(a) the white regions are included in the background class, whereas in image 10(b) the white region is included in the foreground class.

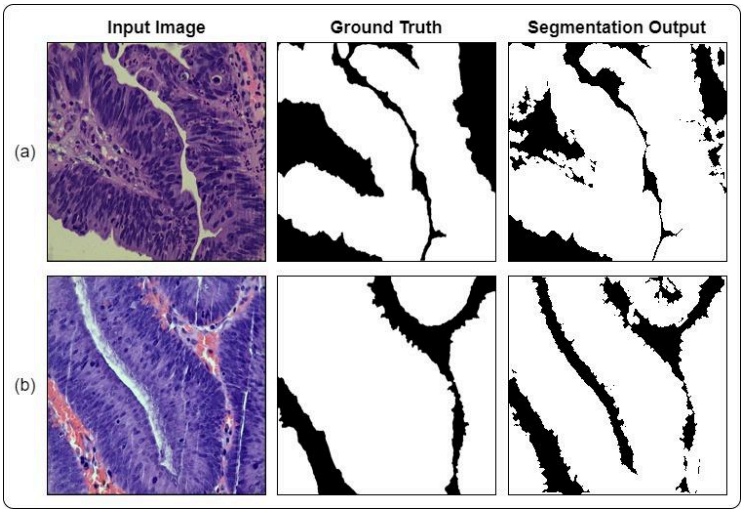


Fig. 10: Segmentation output for two input images of 'High-grade IN' and 'Adenocarcinoma' class type

4.1.7. Results and Discussion

After providing a detailed description about all of our pipelines and also giving an insight into their segmentation quality, we now present the performance evaluation of these pipelines using different evaluation metrics mentioned at the beginning of this section. For this purpose, we have considered 50 images from each class. We have then provided a class-wise segmentation evaluation, followed by the overall evaluation on all classes on 300 images in total; and is displayed in Table 1.

Table 1. Performance Evaluation on all classes individually, followed by the overall evaluation

	Precision (P)	**Recall (R)**	**Jaccard Similarity (J)**	**Dice Score (D)**	**Accuracy (A)**
Normal	0.90	0.78	0.73	0.83	0.81
Polyp	0.89	0.78	0.72	0.83	0.82
Low-grade IN	0.86	0.71	0.63	0.76	0.72
High-grade IN	0.85	0.93	0.80	0.88	0.83
Adenocarcinoma	0.90	0.82	0.74	0.84	0.79
Serrated Adenoma	0.78	0.85	0.67	0.80	0.73
Combined Performance over all classes	0.86	0.81	0.71	0.82	0.78
K-Means [16]	0.62	0.64	0.47	0.62	–

Table 1 shows that the Dice Score values for almost all classes are above 80% and that the Jaccard Similarity scores for those classes are above 70%, which we consider a pretty good performance, given the complexity of the images. Both these metrics calculate the extent of overlap between the output segmentation and the ground truth segmentation. A part of this decreased performance can be attributed to the fact that our method is not detecting the partially visible lumens at the image borders/corner, due to the assumption/requirement we stated previously (about the major portion of a lumen center to be included in the image for it to be segmented by our pipeline). However, an important insight is provided by the Precision and Recall values. The Precision values are high (above 80% for most classes), which shows that the number of true positives w.r.t the total number of positive class outputs is high; whereas the Recall values are a bit lower (above 75% for most classes), that is the number of true positives w.r.t the

actual number of positive values in the ground truth is a bit lower. The credit for these performance values can be given to the Segmentation Refinement Pipeline, which used the watershed algorithm to expand the lumen center. If not used, the lumen borders would not have been segmented properly, leading to an even lower recall rate, and a lower dice and jaccard score (however, in that case the precision would have been even higher because there would have been a lesser number of false positives, leading to an high imbalance between the precision and recall values). However, after using watershed segmentation, there have been some leakages in segmentation, thus it led to an increase in false positives and a decrease in false negatives. This increased the recall, dice and jaccard score, thus preserving the precision-recall trade-off. Further, in the last row of the table, we have presented the best performing image processing based segmentation method used in [16]. It is clear that our proposed pipeline outperformed their methodology by a large margin.

4.2. Machine Learning

In this section, we deal with the task of classifying the input histopathology images into one of the six classes: Normal, Polyp, Low-grade IN, High-grade IN, Adenocarcinoma and Serrated Adenoma. For this purpose, we have employed various image based feature extraction techniques, which are then given to a Machine Learning model for classification. This section is divided into the following subsections: Feature extraction, Classification, and Results and Discussion.

4.2.1. Feature Extraction

For the purpose of feature extraction, we used the following extractors: Gray-Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG) and Gabor filters. With these feature extractors, we have tried to extract the essential underlying information from the image like textural features, orientational features, intensity features, etc. As a preliminary requirement, we have converted all the color images to grayscale images before applying any feature extraction.

Gray-Level Co-occurrence Matrix (GLCM): GLCM is a statistical method for capturing textural features that creates a pixel intensity based co-occurrence matrix, which can be used to calculate different measures. It requires two main parameters: the distance parameter (let it be d) and the angle parameter (let it be θ). At any given time frame, it will capture the texture pattern between two pixels d distance apart, at an angle θ to each other. The co-occurrence matrix contains the total count of these pairwise pixel intensities. Thus, for a more detailed description of the texture feature to be encapsulated, multiple pixel distances, in combination with multiple angles need to be considered. After normalization, this matrix represents the probabilistic distribution of the pairwise pixel intensities.

For our purposes, we used GLCM with multiple values of distance and angles. This will enable us to capture textural information in the image at different angles, distances and scales, thus helping us to achieve a certain level of translational and rotational invariance. In particular, for d we used three values: {5, 10, 15}; and for θ we used four values: $\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$. Also, for achieving some contrast and illumination enhancement, we applied histogram equalization on the images, before applying GLCM to the images for extracting features. From the co-occurrence matrix, we have calculated the following features:

1. **Contrast**: It calculates the intensity difference between the neighboring image pixels.
2. **Correlation**: It calculates the correlation (positive or negative) between the neighboring image pixels.
3. **Energy**: It calculates the randomness in the image texture, that is it measures the un-uniformity in pixel distribution, using the values of the co-occurrence matrix.
4. **Homogeneity**: It gives a measure of the uniformity of the textural patterns in the image, by measuring the closeness of the values in the co-occurrence matrix to the diagonal.

For each distance and angle values, we have four feature values. Texture patterns in each image are represented by three distance values and four angle values, therefore, for each image, we will have (3 x 4 x 4 =) 48 features.

Local Binary Patterns (LBP): LBP is a type of texture descriptor of an image, which captures local patterns in the neighborhood of a pixel. For each pixel, it defines a radius (let r) and the patterns will be captured at the border of the circle of radius r, centered at the pixel. This requires a number of equally separated points (let p) to be defined on the border of the circle and the textures are captured at those points, with respect to the pixel at the center. The higher the number of such points, the more detailed is the texture pattern captured at the radius r.

In our use case, we have used three different LBP feature descriptors: {LBP1: 5, 8}, {LBP2: 15, 10}, and {LBP3: 25, 12}; where the keys represent their respectives codes (used later for referencing) and the values represent the radius of the circles and the number of points used for each LBP descriptor. This will help us achieve scale invariance to some extent. We have used Uniform two-rotation invariant LBP, with which we have achieved rotational invariance and also reduced the number of features to $p+2$ for each LBP descriptor and for each image; because only the pattern occurrence count is kept for each image. Thus, the aggregated count of features for each image is $(8 + 2) + (10 + 2) + (12 + 2) = 36$.

Histogram of Oriented Gradients (HOG): ḤOG features are a set of gradient based feature extractors. Each image is at first divided into cells and then the gradient of each cell is calculated using a 1D derivative Kernel. Then the histogram of oriented histogram is calculated for each cell using the gradient magnitude, binned on the gradient direction, which form the features for each cell. Finally, the cells are grouped into blocks, where each block can contain one or more cells, both row and column wise. Accordingly the cell features are also grouped into the block features. This block-wise grouping is performed for the entire image and the block features are joined together to form the feature vector for each image.

A major issue of this process is that the number of features scale exponentially. A way to reduce the number of features is to use this method only in a small Region of Interest (RoI). However, in case of histopathology images, no specific RoIs can be declared, due to the randomness in the distribution of the lumen and also in their shape and size. Thus, we had to apply this method on the entire image. To keep a limit on the number of features, we used larger cell sizes. Typically, we have used two different sets of HOG features: {HOG1: (16, 16), (1, 1)} and {HOG2: (32, 32), (2, 2)}; where the keys represent their respective codes, the first pair in the values represent the number of pixels per cell and the second pair represents the number of cells per block. We got 1764 features from the first set of HOG features and 1296 features from the second HOG feature set.

Gabor filters: The final set of feature extractors we used are the Gabor filters. They use properties of both spatial (gaussian component) and frequency (sinusoidal waves) domains for capturing features from an input image. Seven parameters are needed in total for defining one convolutional filter, which are:

1. $\boldsymbol{x}$, $\boldsymbol{y}$: size of the gabor filter
2. $\boldsymbol{\gamma}$: ellipticity of the filter
3. $\boldsymbol{\lambda}$: wavelength of the sinusoidal wave
4. $\boldsymbol{\theta}$: orientation of the wave
5. $\boldsymbol{\Psi}$: phase shift of the wave
6. $\boldsymbol{\sigma}$: spatial spread (standard deviation) of the gaussian component

These parameters give adequate control over the filter and thus can help us in capturing a much broader range of textural features. However, at the same time, we are left with too many gabor filters to consider. We have considered the following values for the parameters: {$\boldsymbol{x}$: 21}, {$\boldsymbol{y}$: 21}, {$\boldsymbol{\gamma}$: 0.5}, {$\boldsymbol{\lambda}$: 5, 10, 15}, {$\boldsymbol{\theta}$: 0, $\frac{\pi}{8}$, $\frac{\pi}{4}$, $\frac{3\pi}{8}$, $\frac{\pi}{2}$, $\frac{5\pi}{8}$, $\frac{3\pi}{4}$, $\frac{7\pi}{8}$}, {$\boldsymbol{\Psi}$: 0}, and {$\boldsymbol{\sigma}$: 4}. The output of each filter has the same size as that of the image. We have then extracted the following information from each filter convolution output:

1. **Mean**: It calculates the average of all the intensity values in the output image.
2. **Variance**: It is the spread of the intensity values around the mean.
3. **Energy**: It is the overall strength of the magnitude of the pixel values of the output image.

4. **Kurtosis**: Kurtosis measures the overall deviation of the distribution of the image pixel values from a normal distribution.
5. **Skewness**: It measures the asymmetry in pixel value distribution of the image around the mean.
6. **Contrast**: It measures the overall variation in pixel values between the neighboring pixels.
7. **Homogeneity**: It is a measure of the uniformity of the intensity values distribution in the image.

After considering all the possible parameter combinations, a total of 24 Gabor filters were used. Each gabor filter contributed to those 7 parameters. So, in total, we had (24 x 7 =) 168 feature contributions from each image.

4.2.2. Classification

For the purpose of classification of the images using a combination of the extracted features, we performed some necessary preprocessing on the extracted features, followed by training an appropriate Machine Learning model. The preprocessing steps included are:

1. **Principal Component Analysis (PCA)**: The HOG feature descriptors have returned a huge feature set, which should further be combined with other extracted features, resulting in a further larger feature set. This large feature set can be a problem for us, given our limited availability of histopathology image instances (Curse of Dimensionality). Thus PCA has been implemented for dimensionality reduction.
2. **Train Test Split**: The entire dataset was divided into training and testing datasets. We used 80% data to train the models and 20% data to evaluate the effectiveness of the Machine Learning Algorithms. Thus, the training set has 1782 instances and the testing set has 446 instances.
3. **Standardization**: The extracted features had values spread over a variety of ranges. So standardization was required to bring them down to a similar scale. This is an essential requirement for the better convergence of many Machine Learning algorithms. It is performed as per the following formula:

   $$z_i = \frac{x_i - \mu}{\sigma}$$

   Where, z_i is the new standardized feature corresponding to the input feature x_i;

   $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$ is the mean; and

   $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (x_i - \mu)}$ is the standard deviation

4. **SMOTE**: As mentioned in the Dataset description section, there is a huge imbalance among the different dataset classes. So to balance the classes in the training dataset, Synthetic Minority Oversampling Technique was applied. It creates new data samples for the minority classes by interpolating data samples in between the existing ones using the K-Nearest Neighbour technique. As a result of applying SMOTE, our total number of training data instances increased from 1782 to 3816 instances.

After preprocessing, the following machine learning models were considered for inclusion in the multiclass classification of the histopathology images:

Random Forest: Random Forest (RF) is an ensemble of multiple decision trees, typically using the bagging method, to achieve more accurate predictions. It extends the bagging approach by incorporating both bagging and feature randomness, creating an uncorrelated forest of decision trees. Instead of identifying the most important feature for each node split, it selects the best feature from a random subset of features. Additionally, Random Forest can measure the relative importance of each feature in making predictions. This method effectively reduces the risk of overfitting, as averaging the predictions of uncorrelated trees decreases overall variance and prediction error.

Support Vector Classifier: Support Vector Classifier (SVC) is a specialized implementation of Support Vector Machine (SVM) designed for classification tasks. SVC aims to identify the optimal hyperplane that can distinguish

between different classes of data points. It is capable of handling both linear and non-linear classification problems by employing kernel functions. These functions transform the original feature space into a higher-dimensional space where a linear separation is feasible. Common kernel functions used in SVC include linear, polynomial, and radial basis function (RBF) kernels.

Gradient Boosting: Gradient Boosting Decision Tree (GBDT) is a widely-used machine learning algorithm for classification tasks. It works by combining multiple weak learners to create a strong predictive model. Each weak learner is trained to minimize a chosen loss function, such as mean squared error or cross-entropy, using gradient descent. It is an iterative process that allows Gradient Boosting to effectively improve model accuracy and robustness over time. Additionally, Gradient Boosting is highly flexible and can be adapted to various types of data and problems.

LightGBM: The gradient boosting decision tree (GBDT) often requires considerable time as it evaluates every data instance to determine the information gain for all potential splits of each feature. This process is particularly time consuming, inefficient and challenging when dealing with high dimensional features and large datasets, making it difficult to achieve satisfactory results. LightGBM (LGBM), a new GBDT based model delivers nearly the same performance as traditional GBDT but trains faster. An advantage of LightGBM is its ability to optimally partition categorical features, further enhancing its versatility and efficiency in various machine learning tasks.

XGBoost: XGBoost (XGB), which stands for 'extreme gradient boosting', is a decision tree based ensemble method particularly effective for boosting. It employs a greedy algorithm and rapidly determines optimal parameters through distributed processing. XGboost also has a flexible learning system allowing for model optimization via various adjustable parameters.

The entire Machine Learning pipeline is summarized and visualized in Fig. 11. The hyperparameters set for each ML model is shown in Table 2. After multiple experimentations, these were the best values found.

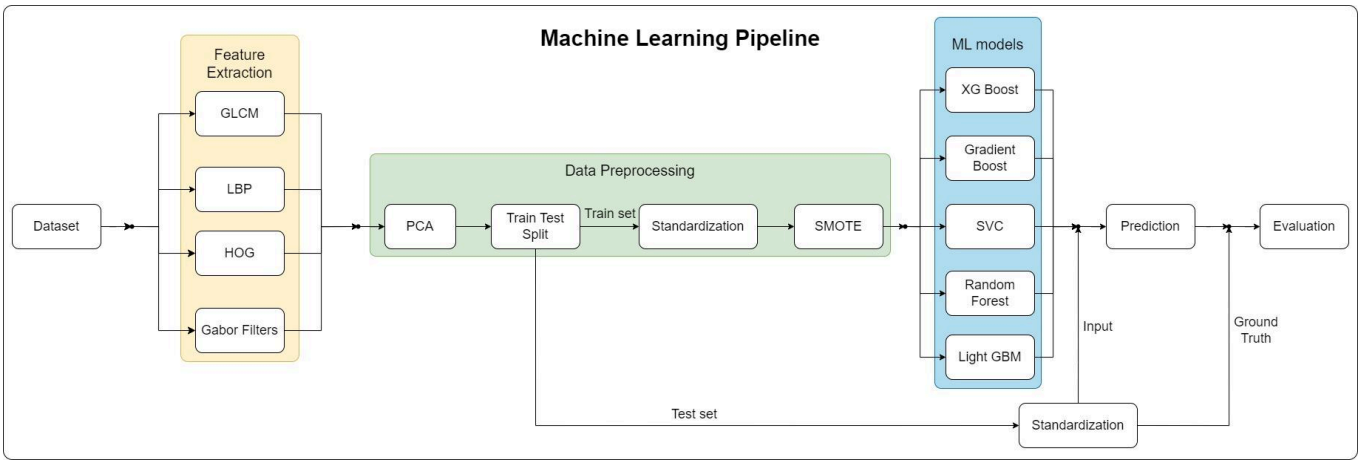


Fig. 11: An overview of the Machine Learning Pipeline

Table 2: Hyperparameter values for each model

Models	**XGB**	**GB**	**SVC**		**RF**	**LGBM**
Hyperparameters	n_estimators	n_estimators	C	kernel	n_estimators	n_estimators
Values	300	300	1	RBF	400	70

4.2.3. Results and Discussion

After giving a detailed insight into our feature extraction method, the preprocessing steps and the Machine Learning models we considered for training, in this section, we present the performance evaluation of our proposed approach. It displays the detailed evaluation of the ML models on different combinations of the extracted features

(which shows a brief insight into our features selection process), thus demonstrating not only the performance of the ML algorithms in classification, but also the importance of the features extracted by each feature extractor, thus determining their contribution (and necessity) towards histopathology images (Table 3 and Table 4).

Table 3: Detailed Performance Evaluation of the ML pipeline, with multiple models trained on features extracted by individual feature extractors

Features	Models	Precision (P)	Recall (R)	Accuracy (A)
LBP1	XGB	0.62	0.63	0.63
	GB	0.61	0.63	0.63
	SVC	0.60	0.66	0.66
	RF	0.65	0.66	0.66
	LGBM	0.60	0.64	0.64
LBP2	XGB	0.65	0.66	0.66
	GB	0.61	0.63	0.63
	SVC	0.67	0.68	0.68
	RF	0.64	0.66	0.66
	LGBM	0.63	0.65	0.65
LBP3	XGB	0.53	0.56	0.56
	GB	0.53	0.54	0.54
	SVC	0.62	0.60	0.60
	RF	0.54	0.57	0.57
	LGBM	0.54	0.57	0.57
GLCM	XGB	0.59	0.62	0.62
	GB	0.57	0.61	0.61
	SVC	0.56	0.60	0.60
	RF	0.55	0.59	0.59
	LGBM	0.56	0.60	0.60
HOG1	XGB	0.34	0.41	0.41
	GB	0.37	0.40	0.40
	SVC	0.33	0.39	0.39
	RF	0.26	0.40	0.40
	LGBM	0.31	0.38	0.38
HOG2	XGB	0.44	0.49	0.49
	GB	0.43	0.47	0.47
	SVC	0.43	0.52	0.52
	RF	0.36	0.43	0.43
	LGBM	0.43	0.50	0.50
Gabor filters	XGB	0.75	0.76	0.76
	GB	0.73	0.74	0.74
	SVC	0.75	0.75	0.75
	RF	0.75	0.73	0.73
	LGBM	0.73	0.74	0.74

Table 3 shows the Precision, Recall and Accuracy of the different models on the features extracted by individual feature extractors and Table 4 shows these metric values for the model performances on different combinations of these features. It can be noted that when no features are combined, the best performance is displayed by the models trained on Gabor features, indicating that it has been most effective in extracting the essential features from the images. Comparable results were obtained when the features from all the LBPs were combined (Table 4). The lowest performance was shown by the models trained on the HOG features (Table 3). This can be attributed to the fact that there are too many cells spread all over the histopathology images, which have a close to circular shape. Inside a 16x16 or a 32x32 cell size, there are too many gradients in all directions due these small circular objects, resulting in no useful gradient information being extracted from a cell/block. The performance of the models were further depleted, when the HOG features were combined with other features (Table 4). These HOG features were included after their dimensionality was reduced to 400 features using PCA. After multiple experimentations, we found that the best set of features were provided by the combination of the three LBP features, the GLCM features

and the Gabor extracted features. However, even after numerous experiments, no significant improvement in performance was seen after the usage of SMOTE on training data for handling data imbalance. As Table 4 shows, the best performance is returned by the XGB model trained on the best set of features, without SMOTE. The Precision, Recall and Accuracy values are also close to each other, indicating the conservation of the Precision-Recall trade-off and a low chance of any additional bias or variance presence in the model.

Table 4: Detailed Performance Evaluation of the ML pipeline, with multiple models trained on different combination of features

Features	Number of Features	Models	Precision (P)	Recall (R)	Accuracy (A)
LBP1 + LBP2 + LBP3	36	XGB GB SVC RF LGBM	0.73 0.72 0.76 0.70 0.75	0.74 0.73 0.77 0.71 0.75	0.74 0.73 0.77 0.71 0.75
LBP1 + LBP2 + LBP3 + HOG1 + PCA (n=400)	436	XGB GB SVC RF LGBM	0.67 0.69 0.56 0.63 0.70	0.71 0.72 0.64 0.64 0.71	0.71 0.72 0.64 0.64 0.71
LBP1 + LBP2 + LBP3 + HOG2 + PCA (n=400)	436	XGB GB SVC RF LGBM	0.70 0.69 0.60 0.61 0.71	0.73 0.72 0.64 0.64 0.73	0.73 0.72 0.64 0.64 0.73
LBP1 + LBP2 + LBP3 + GLCM + Gabor filters	252	**XGB** GB SVC RF LGBM	**0.83** 0.82 0.81 0.77 0.82	**0.83** 0.82 0.81 0.77 0.82	**0.83** 0.82 0.81 0.77 0.82
LBP1 + LBP2 + LBP3 + GLCM + Gabor filters + SMOTE	252	XGB GB SVC RF LGBM	0.82 0.82 0.82 0.78 0.81	0.81 0.82 0.82 0.78 0.81	0.81 0.82 0.82 0.78 0.81

Table 5: Confusion Matrix of XGB model trained on the best combination of features

		Predicted Label					
		Normal	**Polyp**	**Adenocarcinoma**	**High-grade IN**	**Low-grade IN**	**Serrated Adenoma**
Actual Label	**Normal**	8	5	0	0	2	0
	Polyp	1	83	4	0	7	0
	Adenocarcinoma	0	6	151	1	1	0
	High-grade IN	0	0	15	14	8	0
	Low-grade IN	0	9	9	1	108	1
	Serrated Adenoma	0	0	2	1	1	8
	Class-wise Accuracy	0.88	0.81	0.84	0.82	0.85	0.88

After establishing the best set of features and the best model for histopathology image classification, we now present the confusion matrix for this model in Table 5. It clearly displays the class imbalance problem in our dataset ('Normal' class has just 9 test instances, whereas 'Adenocarcinoma' class has 178 test instances). On the good side, it can be seen that all the classes have similar performance values (and all of them being above 80%), indicating that no class is underperformed by the model and no class is getting greater priority than the others.

4.3. Deep Learning

In this section of our colorectal cancer (CRC) study, we perform segmentation of different types of colorectal histopathology images using deep learning techniques. For this purpose, we primarily employed different Unet architectures including the vanilla Unet [17] with VGG16 backbone. We also used two other backbones including ResNet50 and EfficientNet-b2. In the later experiments, we replaced the UNet with the nested UNet++ [18] and also incorporated scSE (Concurrent Spatial and Channel Squeeze & Excitation) attention mechanisms [19], enhancing its ability to focus on relevant features within the images. UNet++ was used because of its added complexity on the decoder part and also its dense skip connections which propagates the features better than the original UNet. All of the models were initialized using ImageNet pretrained weights.

4.3.1. ImageNet Dataset

The ImageNet dataset [20] is a comprehensive visual database containing over 14 million images that are manually annotated across nearly 20,000 categories. Following the pioneering work of Krizhevsky et al. (2017) with AlexNet, which won the ImageNet 2012 Challenge, many deep learning backbones have been trained on ImageNet and are widely available for fine-tuning. In our study, we leveraged these pre-trained weights to initialize our segmentation networks, providing several significant advantages:

1. Faster Convergence: Using pre-trained weights facilitated faster convergence during training, as the models began with already learned features relevant to image recognition. This pre-training helped our models to quickly adapt to our specific task of segmenting colorectal cancer (CRC) histopathology images.

2. Improved Performance: Pre-trained models often achieve improved performance, particularly when the target dataset is limited in size, by transferring learned features from a large and diverse dataset.

3. Enhanced Feature Extraction: The rich set of features learned by these models on the ImageNet dataset is highly generalizable, which is particularly beneficial for the initial layers of the network. This enhances feature extraction and overall model robustness.

4.3.2. Model and Architecture

1. Segmentation Architectures

UNet: UNet is one of the most widely used biomedical image segmentation models, due its simplicity and effectiveness. It has a symmetric encoder-decoder structure, connected through intricate skip connections to preserve spatial information. The encoder progressively captures features at various spatial resolutions, while the decoder reconstructs these features into a precise segmentation mask. This architecture ensures that both the global context and the fine-grained details of the image are maintained, making it suitable and adept at tasks that require detailed delineation of structures within medical images.

UNet++: Built on the solid foundation of UNet, UNet++ introduces a novel approach with its nested and dense skip connections, significantly improving feature propagation and refinement and addressing the semantic gap between the encoder and decoder feature maps. It has a more complex decoder structure that allows for superior feature reuse, resulting in highly accurate segmentation. UNet++ can be particularly advantageous for complex histopathology images, where subtle differences in tissue structures are critical. The added complexity and refined connections in UNet++ may enable it to outperform the original UNet in capturing these intricate details.

2. Backbones

Backbones are pre-trained models that serve as feature extractors of a segmentation architecture. They are typically trained on large and diverse datasets, such as ImageNet, to learn rich feature representations that can be transferred to other tasks.

VGG16: The VGG16 backbone, pre-trained on the extensive ImageNet dataset, is renowned for its straightforward yet powerful architecture. Comprising 13 convolutional layers, VGG16 excels in capturing fine texture details and structural nuances. When integrated into the UNet framework, this backbone enhances the model's ability to segment detailed and intricate regions within histopathology images. The pre-training on ImageNet provides a solid foundation, enabling the network to leverage rich, pre-learned features that expedite convergence and improve overall segmentation performance.

ResNet50: ResNet50 introduces a deeper, more sophisticated architecture with 50 layers, utilizing residual connections to mitigate the vanishing gradient problem commonly encountered in deep networks. This backbone is particularly effective in capturing complex features and patterns, making it well-suited for advanced segmentation tasks. In the context of UNet, the ResNet50 backbone strikes a balance between depth and computational efficiency, allowing the model to handle complex histopathological features with enhanced accuracy and robustness.

EfficientNet B0, B1, B2: EfficientNet architecture employs a compound scaling method to balance depth, width, and resolution efficiently. Starting with EfficientNet-B0, this baseline model is designed for real-time and resource-constrained environments, offering a lightweight yet effective solution for segmentation tasks. Pre-trained on ImageNet, it provides a robust feature set right from the start. Moving up, EfficientNet-B1 increases its capacity, enhancing performance while maintaining efficiency. It is ideal for applications requiring a moderate computational increase, providing a balance that suits many standard hardware setups. EfficientNet-B2 further scales these dimensions, delivering higher accuracy and better performance for large-scale and detailed segmentation tasks. Its pre-training on ImageNet ensures a rich initialization for precise medical image analysis and other detailed segmentation needs. Each model, from B0 to B2, integrates seamlessly into UNet architectures, boosting segmentation precision while staying computationally feasible. This makes the EfficientNet series a state-of-the-art solution for diverse and demanding segmentation challenges.

3. Attention Mechanisms

scSE (Concurrent Spatial and Channel Squeeze & Excitation): The scSE attention mechanism is a powerful tool that significantly enhances the model's ability to focus on the most relevant features within an image. By recalibrating feature maps spatially and channel-wise, scSE ensures that the network can highlight critical regions and suppress irrelevant information. This dual attention mechanism is particularly beneficial in medical imaging tasks, where distinguishing between subtle tissue variations is crucial. Integrating scSE into the UNet++ architecture amplifies its effectiveness, enabling the model to achieve superior segmentation accuracy by concentrating on the most pertinent aspects of the histopathology images. This attention mechanism is instrumental in improving the model's precision, making it exceptionally adept at identifying and segmenting cancerous tissues in colorectal histopathology images.

4.3.3. Experiments

1. **Baseline**: We trained and tested a baseline UNet model with the similar specification mentioned in the reference paper of the dataset that we are working on. For this we used the vanilla UNet with VGG16 as the backbone.

2. **UNet with other backbones:** Later we replaced the VGG16 backbone with ResNet50 and EfficientNet-B2 to check if the performance increases or not.

3. **UNet++ with different backbones:** In our study, we employed UNet++ as an alternative to the traditional UNet architecture, utilizing three different backbones: EfficientNet-B0, EfficientNet-B1, and EfficientNet-B2. Initially, our goal was to compare the performance of UNet++ using the same backbones as in the vanilla UNet models. However, we encountered challenges due to the significantly higher complexity and larger parameter sizes associated with UNet++ when paired with backbones like VGG16 and ResNet50. These backbones, coupled with the intricate architecture of UNet++, made it infeasible to train the models effectively within our computational constraints.

The experiments were conducted both with and without image augmentation to check for the role of data augmentation towards the performance of these models.

4.3.4. Implementation

Training of all the models were trained on NVIDIA Tesla P100 16GB GPU. The whole training and inference phases were built using Pytorch Framework. Pytorch Segmentation Models was used to build and load the models which is an open source platform that provides a collection of semantic segmentation models with pre-trained backbones using Pytorch framework. Mixed precision training, also known as half-precision training, leverages both 16-bit (half-precision) and 32-bit (single-precision) floating-point arithmetic to train neural networks. This technique can significantly enhance the efficiency of training deep learning models, particularly on modern GPUs with dedicated support for mixed precision operations. For these reasons, mixed precision training was utilized during the training phase to achieve stable, efficient, and faster gradient optimization and backpropagation. This was accomplished using PyTorch's Automatic Mixed Precision (AMP) package, which dynamically adjusts the precision of computations to optimize performance while maintaining numerical stability. The experiments were tracked using Weights & Biases for efficient comparison between different models.

4.3.5. Data preprocessing

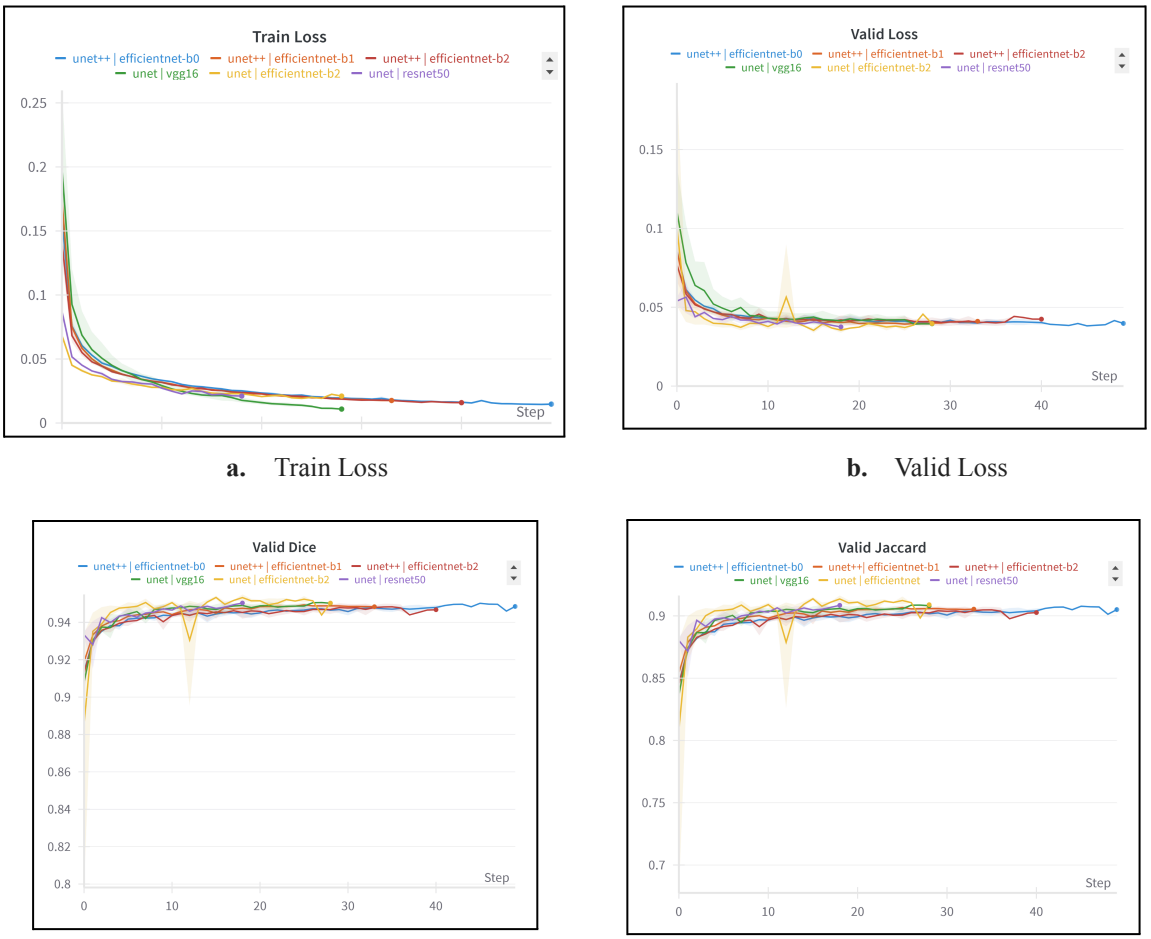
All of the images were of the same size of 224*224 which didn't require any resizing. So the only data preprocessing step was to normalize the images. Since the images were RGB, mean and standard deviation were calculated for the three channels and then the images were normalized across the three channels.

4.3.6. Data Augmentation

Data Augmentation was used in 50% of the training experiments. We used offline-augmentation to increase the number of images in our Dataset. We built a custom sampling function that takes into account the distribution of the cancer types and gives higher weights to minority classes to mitigate the class imbalance in the dataset. Our augmentation pipeline was built to a randomly apply one of the following transformations: (i) Color Jitter with magnitudes (brightness=0.4, contrast=0.4, saturation=0.4), (ii) Gaussian Blur with magnitudes (kernel_size=[3, 7], sigma=[0.1, 3.]), (iii) Random Sharpness Adjustment with magnitudes (sharpness_factor=2) and (v) Gaussian Noise Addition with magnitudes (stddev=0.1).

4.3.7. Training hyperparameters

The model was trained with a batch size of 16 for training and 200 for validation, with an input image size of 224x224 pixels. The training process spanned 50 epochs with an initial learning rate of $1*e^{-4}$, gradually reduced using a Cosine Annealing scheduler to a minimum learning rate of $1*e^{-6}$ which again increased in a cyclic way. The optimization was handled by the Adam optimizer with a weight decay of $1*e^{-5}$. An upgraded Diceloss was used as a loss function to train the model. This particular implementation modifies the standard Dice loss by incorporating squared terms in the denominator, which can help in situations with imbalanced classes. To complement the loss, Dice-coefficient was the primary metric to monitor the training and validation and control certain functionalities such as early stopping criteria during the training phase (for halting training in case of no further performance improvement after 7 epochs). A 5-fold cross-validation approach ensured robustness and generalizability of the model, leveraging a dynamic learning rate scheduler and early stopping to avoid overfitting.



a. Train Loss **b.** Valid Loss

c. Valid Dice Score **d.** Valid Jaccard Similarity

Fig. 12: Performance of UNet with different backbones in Training and Validation Phase.

4.3.8. Performance and Result Analysis

To evaluate our approach, we calculated several metrics including Dice Score, Jaccard Similarity, Precision, Recall, and Accuracy for the models on a dedicated test set. We compared the performance both with and without data augmentation using cross-validation. The following sections have the detailed result analysis based on the approaches and classes.

Training and Validation Performance Evaluation: As shown in Fig. 12, the performance of the three UNet models with different backbones was strong overall, with the EfficientNet-based model slightly outperforming the others with an average Dice Score of 0.954. The Jaccard Similarity scores were slightly lower than the Dice Scores, averaging around 0.9 for all models.

The performance of UNet++ models was also similar to that of the UNet models, albeit with a marginally lower average Dice Score of 0.945 and a Jaccard Score of 0.89, as depicted in Fig. 12. All models employed early stopping criteria and ceased training before reaching 50 epochs to mitigate the risk of overfitting.

Among the models, the vanilla UNet demonstrated better performance compared to the UNet++ variants, and the UNet with the EfficientNet-B2 backbone provided the best results. Notably, the validation loss plateaued at around 0.041, while the training loss plateaued at approximately 0.02, indicating the onset of overfitting in the later

stages of training. Consequently, we saved the model at the epoch where it achieved the best validation score to ensure optimal performance.

We also conducted experiments incorporating various image augmentations to evaluate their effect on model performance. Contrary to our expectations, the application of augmentations did not lead to any improvements; instead, there was a slight decrease in performance across all models. This decline in performance can be attributed to the unique characteristics of histopathology images. Histopathology images exhibit significant heterogeneity in texture and shape, which is critical for accurate segmentation. However, the color of the stains used in these images is highly specific and standardized. Augmenting images by altering contrast, brightness, or applying other transformations can inadvertently distort these color-specific features, leading to a loss of crucial information that the models rely on for accurate segmentation.

For instance, adjustments in contrast or color balance might obscure subtle yet important differences in tissue types that are essential for distinguishing between healthy and cancerous cells. This alteration likely explains the reduction in segmentation accuracy observed after applying augmentations. For this reason we decided to use models trained on the original dataset without augmentations for the inference phase. The decision is rooted in the unique properties of histopathology images, where color consistency and specific staining patterns are crucial for accurate segmentation.

Inference Performance Evaluation: In our evaluation, we assessed the inference performance of various UNet and UNet++ models with different backbones, focusing on the previously mentioned key metrics. The results, detailed in Table 6, highlight the comparative performance of these models. The baseline UNet with VGG16 backbone achieved a Dice Score of 0.9337 and a Jaccard Similarity of 0.8921, indicating strong segmentation performance. This model also demonstrated high Precision (0.9305), Recall (0.9397), and Accuracy (0.9402). The UNet with ResNet50 backbone showed slightly improved performance, with a Dice Score of 0.9353 and a Jaccard Similarity of 0.8952. This configuration also maintained high Precision (0.9308) and Recall (0.9429), resulting in an Accuracy of 0.9421. Among the UNet models, the UNet with EfficientNet-B2 backbone outperformed the others, achieving the highest Dice Score of 0.9372 and a Jaccard Similarity of 0.8986. It also excelled in Precision (0.9323), Recall (0.9448), and Accuracy (0.9448), showcasing its superior ability to balance performance and efficiency.

Table 6: Detailed Performance of DL models with different combinations in Inference Phase

Models	Dice Score	Jaccard Similarity	Precision (P)	Recall (R)	Accuracy (A)
Baseline Unet Backbone: VGG16	0.9337	0.8921	0.9305	0.9397	0.9402
UNet Backbone: ResNet50	0.9353	0.8952	0.9308	0.9429	0.9421
UNet Backbone: EfficientNet-B2	**0.9372**	**0.8986**	**0.9323**	**0.9448**	**0.9448**
UNet++ Attention: scSE Backbone: EfficientNet-B0	0.9328	0.8909	0.9315	0.9377	0.9387
UNet++ Attention: scSE Backbone: EfficientNet-B1	0.9337	0.8926	0.9305	0.9408	0.9403
UNet++ Attention: scSE Backbone: EfficientNet-B2	0.9289	0.8846	0.9211	0.9417	0.9330

For the UNet++ models, UNet++ with EfficientNet-B0 and scSE attention delivered a Dice Score of 0.9328 and a Jaccard Similarity of 0.8909. Despite slightly lower metrics, it maintained high Precision (0.9315) and Recall

(0.9377), with an Accuracy of 0.9387. The UNet++ with EfficientNet-B1 and scSE attention performed similarly to the B0 variant, with a Dice Score of 0.9337 and a Jaccard Similarity of 0.8926. It also exhibited robust Precision (0.9305), Recall (0.9408), and Accuracy (0.9403). However, the UNet++ with EfficientNet-B2 and scSE attention had a lower performance compared to the other UNet++ configurations, with a Dice Score of 0.9289 and a Jaccard Similarity of 0.8846. Its Precision (0.9211) and Accuracy (0.9330) were also somewhat reduced, although Recall (0.9417) remained high.

Overall, the analysis indicates that while UNet models with EfficientNet-B2 backbone consistently outperformed others, the UNet++ variants, particularly those with EfficientNet-B1, provided competitive results.

Class-wise Inference Performance Evaluation: During inference, the class-wise performance of our model closely mirrored the results observed during validation. Table 7 provides a detailed breakdown of the model's performance across various tissue classes, highlighting its capability in segmenting different types of tissues. **Normal tissues** posed a greater challenge for the model, achieving a Dice Score of 0.6267 and a Jaccard Similarity of 0.6062. Despite these lower scores compared to other tissue types, the model maintained high Precision (0.6236), Recall (0.6279), and Accuracy (0.9745), indicating that the segmentation was still reasonably reliable.

For **Adenocarcinoma**, the model recorded a Dice Score of 0.9323 and a Jaccard Similarity of 0.8778. The Precision and Recall were 0.9292 and 0.9415, respectively, with an Accuracy of 0.9188. These scores highlight the model's competence in segmenting malignant tissues, which is crucial for effective cancer diagnosis. In segmenting **Serrated Adenoma**, the model delivered a Dice Score of 0.9459 and a Jaccard Similarity of 0.8993. The performance remained robust with a Precision of 0.9482, Recall of 0.9471, and an Accuracy of 0.9303, indicating the model's effectiveness in identifying these specific adenomas.

Table 7: Detailed performance of the best model with different evaluation metrics for each class followed by the combined performance for all classes.

	Dice Score	Jaccard Similarity	Precision	Recall	Accuracy
Normal	0.6267	0.6062	0.6236	0.6279	0.9745
Polyp	0.9710	0.9442	0.9656	0.9771	0.9682
Low-grade IN	0.9646	0.9458	0.9646	0.9801	0.9625
High-grade IN	0.9391	0.8868	0.9331	0.9479	0.9221
Adenocarcinoma	0.9323	0.8778	0.9292	0.9415	0.9188
Serrated Adenoma	0.9459	0.8993	0.9482	0.9471	0.9303
Combined Performance over all classes	0.9372	0.8986	0.9323	0.9448	0.9448

The model excelled in segmenting **Polyp tissues**, achieving a remarkable Dice Score of 0.9710 and a Jaccard Similarity of 0.9442. These metrics reflect the model's ability to accurately delineate polyps, with Precision and Recall scores of 0.9656 and 0.9771, respectively, and an overall Accuracy of 0.9682. This high performance suggests the model's robustness in detecting and segmenting polyps, which are critical for identifying early-stage abnormalities. For both **Low-grade IN (Intraepithelial Neoplasia)** and **High-grade IN**, the model performed exceptionally well, with high performance metric values, thus demonstrating the model's effectiveness in segmenting these cancer stages accurately.

Overall, the combined performance across all classes was strong, with a high average Dice Score and Jaccard Similarity values. These metrics underscore the model's reliable performance in segmenting a variety of tissue types, particularly excelling in detecting and delineating cancerous tissues. The consistent performance across different classes demonstrates the model's robustness and potential for practical usage in medical image analysis.

Visual Analysis of Segmentation Performance: Fig. 13 provides a comparative visual representation of the segmentation masks generated by various UNet models with different backbones across different cancer types. These models include the baseline UNet with VGG16, UNet with ResNet50, and UNet with EfficientNet variants (B0, B1, B2).

Across the images, we observe that the models generally produce similar segmentation outputs, particularly in distinguishing the cancerous tissues from the background. However, they encounter challenges in accurately segmenting certain classes, especially when dealing with subtle or intricate pixel variations. For instance, in the segmentation of **Low-grade IN (Intraepithelial Neoplasia)**, all models show consistent discrepancies when compared to the ground truth masks. These differences highlight the models' difficulties in precisely capturing the fine details of this class, suggesting a need for further refinement or additional post-processing to improve accuracy. **Polyp** tissues are well-segmented across all models, reflecting the high Dice Score and Jaccard Similarity metrics reported earlier. This consistency underscores the models' robustness in handling more distinct and less ambiguous tissue structures.

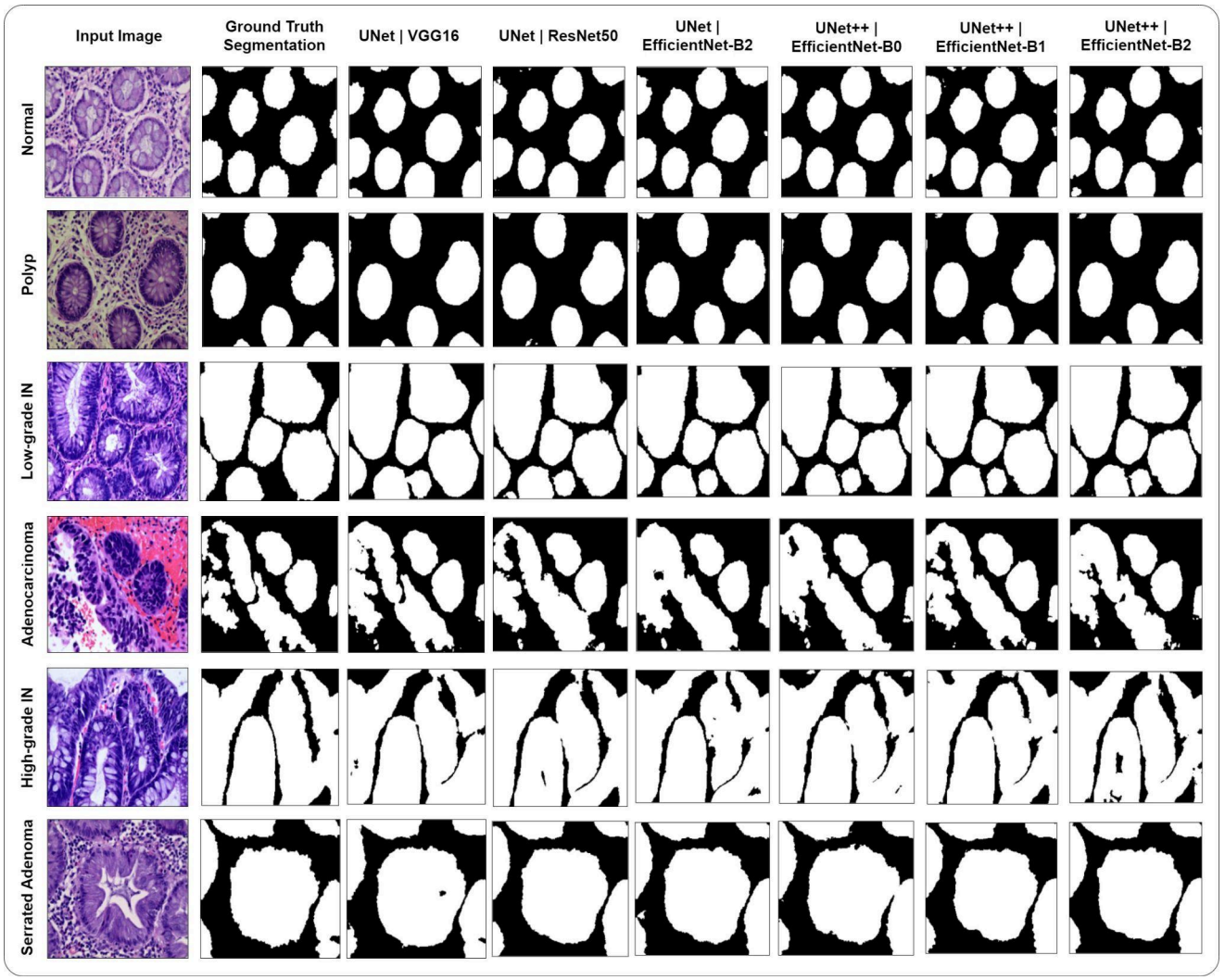


Fig. 13: Predicted Segmentation Masks by UNet with different backbones for all the cancer types.

For **High-grade IN** and **Adenocarcinoma**, the models produce segmentation masks that are closely aligned with the ground truth, though slight differences are still observable.This could be attributed to the tissues' more heterogeneous nature, making them harder to segment accurately. These slight variations emphasize the need for models that can better capture the complexity and variability within these cancer types. **Serrated Adenoma** segmentation is another area where models perform well, yet subtle segmentation differences still appear. These differences may impact the precision required for clinical applications, highlighting the importance of further tuning and optimization.

Overall, the visual inspection in Fig. 13 reveals that while the UNet models with different backbones perform effectively in segmenting various cancerous tissues, there remain consistent challenges across all classes, particularly in capturing fine details. The consistent patterns of segmentation discrepancies suggest that enhancing model sensitivity to pixel-level variations could be beneficial in improving segmentation quality across all classes. This analysis supports the quantitative metrics presented earlier, illustrating the strengths and weaknesses of each model visually and underscoring the need for ongoing improvements in medical image segmentation tasks.

The original dataset contains the patch size images instead of the whole slide hence the shape of the tissues are not intact in the corners which eventually leads to difficulty in the segmentation. An overview from Fig. 13 gives the insight that the models are generally having difficulty in segmenting the tissues in the corners for every image.

5. Conclusion

In this work, we addressed the issue of Colorectal Cancer classification and segmentation from histopathological images, and we accordingly proposed three different solutions. The first solution is purely based on advanced image analysis techniques, which is used to perform image segmentation. We accordingly designed three specialized pipelines for the different classes of the histopathological images. The average dice score for this solution is 82%. The next solution is for the classification of the images into one of the six types: Normal, Polyp, Low-grade IN, High-grade IN, Serrated Adenoma and Adenocarcinoma. The solution is based on extracting features from the images using Linear Binary Patterns, Gray-Level Co-occurrence Matrix and Gabor filters, and then using those features for training multiple classification models like XGBoost, SVC, etc and selecting the best performing model out of them. The classification accuracy is 83% by XGBoost model. The final solution is based on deep learning, and has been employed to perform segmentation of the histopathology images. After extensive experimentations, the UNet model with an EfficientNet-B2 backbone was found to be the best performing model with a dice score of 93%, thus clearly demonstrating its capability in extracting semantically essential features from the input image and performing an accurate segmentation.

Possible future works can include developing more efficient pipelines and models that can help in achieving an even higher performance, like using an ensemble of models. Other future works can include checking the effectiveness of the solutions proposed for colorectal cancer histopathology images on other types of histopathology images.

6. References

1. Fenoglio-Preiser CM, Hutter RV. Colorectal polyps: pathologic diagnosis and clinical significance. CA Cancer J Clin. 1985;35(6):322-344.
2. Gurcan, M. N., Boucheron, L. E., Can, A., Madabhushi, A., Rajpoot, N. M., & Yener, B. (2009). Histopathological image analysis: a review. IEEE reviews in biomedical engineering, 2, 147–171.
3. Pietikainen M., Hadid A., Zhao G., Ahonen T. Computer Vision Using Local Binary Patterns. Volume 40 Springer; London, UK: 2011.
4. Rathore, S., & Iftikhar, M.A. (2016). CBISC: A Novel Approach for Colon Biopsy Image Segmentation and Classification. Arabian Journal for Science and Engineering, 41, 5061 - 5076.
5. Mármol, I., Sánchez-de-Diego, C., Pradilla Dieste, A., Cerrada, E., & Rodriguez Yoldi, M. J. (2017). Colorectal Carcinoma: A General Overview and Future Perspectives in Colorectal Cancer. International journal of molecular sciences, 18(1), 197.
6. Chaddad, A.; Tanougast, C. Texture Analysis of Abnormal Cell Images for Predicting the Continuum of Colorectal Cancer. Anal. Cell Pathol. 2017,2017, 8428102
7. Cao, H., Bernard, S., Heutte, L., & Sabourin, R. (2018). Improve the performance of transfer learning without fine-tuning using dissimilarity-based multi-view learning for breast cancer histology images. International Conference on Image Analysis and Recognition.
8. Rathore, S., Iftikhar, M. A., Chaddad, A., Niazi, T., Karasic, T., & Bilello, M. (2019). Segmentation and Grade Prediction of Colon Cancer Digital Pathology Images Across Multiple Institutions. Cancers, 11(11), 1700.

9. Kurmi, Yashwant & Chaurasia, Vijayshri & Ganesh, Narayanan. (2019). Tumor Malignancy Detection Using Histopathology Imaging. Journal of Medical Imaging and Radiation Sciences. 50. 10.1016/j.jmir.2019.07.004.
10. Gupta V, Vasudev M, Doegar A, Sambyal N. Breast cancer detection from histopathology images using modified residual neural networks. Biocybernetics Biomed Eng. (2021) 41:1272–87
11. Babu, T., Singh, T., Gupta, D., & Hameed, S. (2022). Optimized cancer detection on various magnified histopathological colon images based on DWT features and FCM clustering. *Turkish J. Electr. Eng. Comput. Sci., 30*, 1-17.
12. A. Ben Hamida, M. Devanne, J. Weber, C. Truntzer, V. Derangère, F. Ghiringhelli, G. Forestier, C. Wemmert (2022). Weakly Supervised Learning using Attention gates for colon cancer histopathological image segmentation. Artificial Intelligence in Medicine, Volume 133.
13. Talukder, M., Islam, M.M., Uddin, M.A., Akhter, A., Hasan, K.F., & Moni, M.A. (2022). Machine Learning-based Lung and Colon Cancer Detection using Deep Feature Extraction and Ensemble Learning. Expert Syst. Appl., 205, 117695.
14. Tharwat, M., Sakr, N. A., El-Sappagh, S., Soliman, H., Kwak, K. S., & Elmogy, M. (2022). Colon Cancer Diagnosis Based on Machine Learning and Deep Learning: Modalities and Analysis Techniques. Sensors (Basel, Switzerland), 22(23), 9250.
15. Sakr AS, Soliman NF, Al-Gaashani MS, Pławiak P, Ateya AA, Hammad M. An Efficient Deep Learning Approach for Colon Cancer Detection. Applied Sciences. 2022; 12(17):8450.
16. Shi L, Li X, Hu W, et al. EBHI-Seg: A novel enteroscope biopsy histopathological hematoxylin and eosin image dataset for image segmentation tasks. Front Med (Lausanne). 2023.
17. Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18. Springer International Publishing, 2015.
18. Zhou, Zongwei, et al. "Unet++: A nested u-net architecture for medical image segmentation." Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4. Springer International Publishing, 2018.
19. Roy, Abhijit Guha, Nassir Navab, and Christian Wachinger. "Concurrent spatial and channel ‘squeeze & excitation in fully convolutional networks." Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I. Springer International Publishing, 2018.
20. Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.