A Generalized Deep Learning Framework for Whole-Slide Image Segmentation and Analysis

Mahendra Khened*[a], Avinash Kori*[a], Haran Rajkumar*[a], Balaji Srinivasan[b], Ganapathy Krishnamurthi[a,]

[a] *Department of Engineering Design*
Indian Institute of Technology Madras, Chennai, India,
* *These authors contributed equally to this work.*
[b] *Department of Mechanical Engineering*
Indian Institute of Technology Madras, Chennai, India

Abstract

Histopathology tissue analysis is considered the gold standard in cancer diagnosis and prognosis. Whole slide imaging, i.e., the scanning and digitization of entire histology slides, are now being adopted across the world in pathology labs. Trained histopathologists can provide an accurate diagnosis of biopsy specimens based on whole slide images (WSI). However, given the large size of these images and the increase in the number of potential cancer cases, an automated solution as an aid to histopathologists is highly desirable. In the recent past, deep learning-based techniques, namely, CNNs, have provided state of the art results in a wide variety of image analysis tasks, including analysis of digitized slides. However, the size of images and variability in histopathology tasks makes it a challenge to develop an integrated framework for histopathology image analysis. We propose a deep learning-based framework for histopathology tissue analysis. We demonstrate the generalizability of our framework, including training and inference, on several open-source datasets, which include CAMELYON (breast cancer metastases), DigestPath (colon cancer), and PAIP (liver cancer) datasets. Our segmentation pipeline is an ensemble of DenseNet, InceptionResnet-V2, and DeeplabV3Plus, where all the networks for each task were trained end to end. We experimentally demonstrate the efficacy of trans-

Email address: gankrish@iitm.ac.in (Ganapathy Krishnamurthi)

ferring model learning from one cancer site to another. We discuss multiple types of uncertainties pertaining to data and model, namely aleatoric and epistemic, respectively. Simultaneously, we demonstrate our model generalization across different data distribution by evaluating some samples on TCGA data. Our framework provides segmentation maps along with uncertainty maps for a given WSI. On CAMELYON16 test data (n=139) for the task of lesion detection, the FROC score achieved was 0.86 and in the CAMELYON17 test-data (n=500) for the task of pN-staging the Cohen's kappa score achieved was 0.9090 (third in the open leaderboard). On DigestPath test data (n=212) for the task of tumor segmentation, a Dice score of 0.782 was achieved (fourth in the challenge). On PAIP test data (n=40) for the task of viable tumor segmentation, a Jaccard Index of 0.75 (third in the challenge) was achieved, and for viable tumor burden, a score of 0.633 was achieved (second in the challenge). Our entire framework and related documentation are freely available at GitHub and PyPi.

Keywords: Whole slide image, Deep learning, Segmentation, Biomedical imaging, Tumor burden, Convex hull, Fully convolutional neural networks, Breast cancer, Lymph node metastases, Liver cancer, Colon cancer, pN-Staging

1. Introduction

Histopathology is still the only definitive method to diagnose cancer (Salamat, 2010). Early diagnosis of cancer significantly increases the probability of survival (Hawkes, 2019). Unfortunately, pathological analysis is an arduous process that is difficult, time-consuming, and requires in-depth knowledge. A study conducted by (Elmore et al., 2015) investigated the concordance of pathologists investigating biopsies of the breast. This study comprised of 115 pathologists across the United States and 240 biopsy specimens. It was found that pathologists disagreed with each other on a diagnosis 24.7% of the time on average. This high rate of misdiagnosis stresses the need to develop computer-

aided methods for histopathology analysis to aid the pathologists. With the increasing prevalence of whole-slide imaging(WSI) scanners that can scan the entire tissue sample, in-silico methods of conducting pathology analysis can now be explored (Madabhushi and Lee, 2016).

Broadly, histopathology analysis includes two types, namely (1) Classification analysis and (2) Segmentation analysis. Classification analysis helps in classifying WSI into benign or malignant (Nanthagopal and Rajamony, 2013), (Guray and Sahin, 2006). Classification can also be used to study cancer subtypes directly (Malhotra et al., 2010). Segmentation analysis helps in detecting and separating tumor cells from the normal cells (Wählby et al., 2004), (Xu et al., 2016). This analysis can also be used for the aforementioned classification task and other analyses like pN-staging and tumor burden estimation.

Automatic histopatholgy analysis is plagued by a myriad of challenges (Tizhoosh and Pantanowitz, 2018);

1. Insufficient training samples: As the data extraction process for medical images is expensive and less frequent, this results in class imbalance issues, adding an inherent skewness to the dataset. This makes it hard to adapt algorithms which are developed for natural images to medical data.
2. Large dimensionality of WSI: A WSI is generated by digitizing a glass slide at a very high resolution of order 0.25 micrometers/pixel (which corresponds to 40X magnification on a microscope). A typical glass-slide of size 20mm x 15mm results in gigapixel image of size 80,000 x 60,000 pixels.
3. Stain variability across laboratories: As the data is acquired from multiple sources, there exists a lack of uniformity in staining protocol. Building a generalized framework that is invariant to stain pattern variability is challenging.
4. Extraction of clinically relevant features and information. Another major challenge is trying to extract features that are meaningful from a clinical point of view. Deep learning does an excellent task of automatic fea-

ture extraction, but understanding these extracted features and extracting meaningful information from them is challenging.

The organization of the paper is as follows. Prior work on histopathology tissues using deep learning methods are discussed in the section 1.1, Followed by our contributions in section 1.2. In section 2.2 we introduce the datasets used in this work, all the preprocessing techniques are discussed in section 2.3. In section 2.4, we discuss multiple deep convolutional networks used, followed by training and inference pipelines in section 2.7 and 2.8 respectively. In section 2.9, 2.10, and 2.6 we describe various other concepts like pN-staging, tumor burden, and uncertainty analysis. Experimental analysis and comprehensive results are presented in section 3 and 4. In section 5 we describe our open-source project (DigiPath AI) and in section 6 we discuss our results and conclude with future direction of work.

1.1. Related work

1.1.1. Deep learning methods for histopathology image analysis

Deep learning methods have shown great success in various tasks involving image, graph, and text data. In the field of medical imaging and diagnosis, deep learning models have achieved human-like results on many specific problems (Kermany et al., 2018), (Bakas et al., 2018), (Weng et al., 2017), (Rajpurkar et al., 2017). One possible medical application where deep learning could have a very high impact if leveraged accurately would be the automation of histopathology analysis. In recent years, grand challenges (Li et al., 2019; CAM, 2017a,b; PAIP, 2019; ICAIR, 2019) are encouraging all the researchers to collectively work on histopathological data using deep learning based solutions, by providing labeled data.

Histopathology slides provide a more comprehensive view of diseases on the tissue and is still considered as a gold standard for cancer diagnosis (Gurcan et al., 2009). In recent past many deep learning-based methods have approached problems of nuclei segmentation (Sirinukunwattana et al., 2016), liver tumor segmentation (Kaluva et al., 2018), epithelial tumor tissue segmentation (Shapcott

et al., 2019), Gleason grading (Arvaniti et al., 2018), signet ringcell detection (Li et al., 2019) and many more. Deep learning-based methods are increasingly used in the context of histopathology analysis because of their ability to automatically discover the representations needed for feature detection from raw data. Deep learning has also helped in merging multi-domain information (Bagari et al., 2018) in diagnosis, since they learn to associate important features from each domain, and has been proven to provide better results.

1.1.2. Lymph node metastases of breast cancer

Breast cancer is one of the most common cancer among women in the world. The prognosis of breast cancer patients is mainly based on the extent of metastases (Sites, 2014). Metastases refers to the spreading of cancer to different parts of the body from where it originally started. This usually occurs when cancer cells break away from the main tumor and enter the bloodstream or lymphatic system. A formally accepted way to classify the extent of cancer is based on TNM staging criteria (Amin and Edge, 2017; Sobin et al., 2011). The TNM staging system takes into account the size of the tumor (T-stage), cancer spreading to regional lymph nodes (N-stage) and metastasization of tumor to other parts of the body (M-stage). In case of breast cancer, the lymph nodes are usually the first location to get metastasized. With the help of sentinel lymph node procedure (Giuliano et al., 2011, 2017), the most likely metastasized lymph nodes are excised and taken for further histopathological processing and examination by a pathologist. The excised node is preserved by fixing in formalin and embedded in paraffin wax block to enable cutting micrometers thin slices of the tissue. These tissue slices are placed on glass slides and are stained with hematoxylin and eosin (H&E). This staining enables the pathologist to differentiate between nuclear (blue) and cytoplasmic parts (pink) of the cell, thereby providing a general overview of the tissue's structure.

Clinically, metastases is divided into one of the three categories namely:- isolated tumor cells (ITC), micro-metastases and macro-metastases. The categorization is based on the diameter of the largest tumor cells cluster. The

Table 1: Size criteria for assigning metastasis type

Category	**Size**
Isolated tumor cells	Single tumor cells or a cluster of tumor cells not larger than 0.2 mm or less than 200 cells
Micro-metastasis	Larger than 0.2 mm and/or containing more than 200 cells, but not larger than 2 mm
Macro-metastasis	Larger than 2 mm

Table 1 provides the size criteria for metastases type. The assigning of pathologic N-stages (pN-stages) is based on metastases size and number of lymph nodes per patient (Sobin et al., 2011). A simplified version of the pN-staging scheme used in CAMELYON17 Challenge (Bejnordi et al., 2017) is provided in Table 2. However, this diagnostic procedure in assessing lymph nodes status is challenging for pathologists as large tissue area has to be examined in detail under microscope at several magnification levels for identifying metastases. Also, this process is tedious, prone to missing small metastases and would require extensive time by pathologist for a thorough examination.

The advent of whole-slide imaging scanners has enabled digitization of glass-slides at very high resolution. Typical whole-slide images (WSI) are in order of gigapixels and usually stored in multi-resolution pyramidal format. These WSIs are suitable for developing computer aided diagnosis systems for automating the pathologist workflow and also with the availability of large amount of data makes WSIs amenable for analysis with machine learning algorithms. Some of the earliest works (Wolberg et al., 1994; Diamond et al., 2004; Petushi et al., 2006) based on machine learning algorithms for digital pathology and WSIs were cancer classification.

Recently, with advent of convolutional neural networks (CNNs) considerable

Table 2: Pathologic lymph node classification (pN-stage) in CAMELYON17 Challenge

pN-Stage	**Slide Labels**
pN0	No micro-metastases or macro-metastases or ITCs found.
pN0(i+)	Only ITCs found.
pN1mi	Micro-metastases found, but no macro-metastases found.
pN1	Metastases found in 1-3 lymph nodes, of which at least one is a macro-metastasis.
pN2	Metastases found in 4-9 lymph nodes, of which at least one is a macro-metastasis.

improvements have been shown in various computer vision tasks like detection, segmentation and classification. CNNs have also been proposed in lymph node metastasis detection in the recent years (Litjens et al., 2016; Paeng et al., 2017; Liu et al., 2017; Lee and Paeng, 2018; Li and Ping, 2018; Wang et al., 2018).

1.1.3. Colorectal carcinoma

Colorectal carcinoma is third most common cancer in the United States (Fleming et al., 2012). Majority of colorectal carcinoma are adenocarcinomas originating from epithelial cells (Hamilton, 2000). Shapcott et al. (2019) discuss the application of deep learning methods for cell identification on TCGA data. Kather et al. (2019) discuss the deep learning methods to predict the clinical course of colorectal cancer patients based on histopathological images and Bychkov et al. (2018) discuss the use of LSTM (Greff et al., 2016) for estimating the patient risk score using spatial sequential memory.

1.1.4. Liver cancer

Three out of every four cases of liver cancer results in death, making it one of the deadliest forms of cancer (Stewart et al., 2019). Amongst the different forms of liver cancer, hepatocellular carcinoma is the most common form(Chuang et al., 2009). Computer-aided methods for detecting liver cancer from vari-

ous modalities such as CT (Li et al. (2015), Yasaka et al. (2017)), ultrasound (Wu et al., 2014) and laparoscopy (Gibson et al., 2017) have been explored. Methods utilizing multi-omics data such as RNA sequencing have also been successful (Chaudhary et al., 2018). However, for histopathology, lack of annotated datasets comprising of liver tissue images in this modality has rendered this area unexplored. The new grand challenge PAIP (2019) on liver cancer segmentation motivates research in this area.

1.2. Contributions

In this paper, we developed an end-to-end framework for histopathology analysis from WSI images that generalizes well with various cancer sites. The framework comprises of segmentation at its core along with novel algorithms that utilize the segmentation to do pathological analyses such as metastasis classification, viable tumor burden estimation, and pN-staging. As discussed in section 1, challenges in WSI analysis are mainly due to their extremely large size, variability in staining, and the limited amount of data. In this work, we propose a few key contributions which helped in addressing a few of the aforementioned problems.

- Extremely large size of WSI: we propose an efficient patch-based approach for training and inference, where sampling is done on a generated tissue mask, thereby reducing computations on unnecessary patches.
- Insufficient data: To address the problems of limited data and class imbalance, we made use of overlapping and oversampling based strategies in patch extraction along with multiple data augmentations.
- Inference Pipeline: To remove the edge artifacts characteristic of patch-based segmentation, we have proposed an overlapping strategy, which proved to reduce them significantly.
- pN Staging: We combined and experimented on various strategies of over-sampling and under-sampling techniques to address class imbalance problem in the dataset for metastases classification.

- Whole tumor segmentation: We propose an empirical non-deep learning based approach for estimating the whole tumor region from the viable tumor region that is computationally efficient yet performs on par with the deep learning based methods.
- Uncertainty Estimation: We have proposed an efficient patch-based uncertainty estimation framework, to estimate both data specific and model (parameter) specific uncertainties.
- Transfer Learning: Based on our study, we show that transfer learning using CAMELYON dataset helps in faster convergence for other histopathology datasets.
- Generalizability: We bench-marked the performance of our methods by validating it on multiple open-source datasets including CAMELYON (Litjens et al., 2018), (Bejnordi et al., 2017), (Bandi et al., 2018) DigestPath(Li et al., 2019), and PAIP(PAIP, 2019).
- Open-source Packaging: Finally, we packaged the framework into an open-source GUI application for the benefit of researchers.

2. Materials and methods

2.1. Overview

The Figure 1, illustrates the proposed overall framework. The proposed framework predicts segmentation mask along with uncertainty maps for a given input image (liver or breast or colon tissue). In this framework we also define a pipeline for pN staging, metastases classification, and tumor burden estimation.

2.2. Data

We made use of multiple open-source datasets including DigestPath19(Li et al., 2019), PAIP19 (PAIP, 2019), CAMELYON(Bejnordi et al., 2017; Bandi et al., 2018; Litjens et al., 2018) datasets to validate our framework.

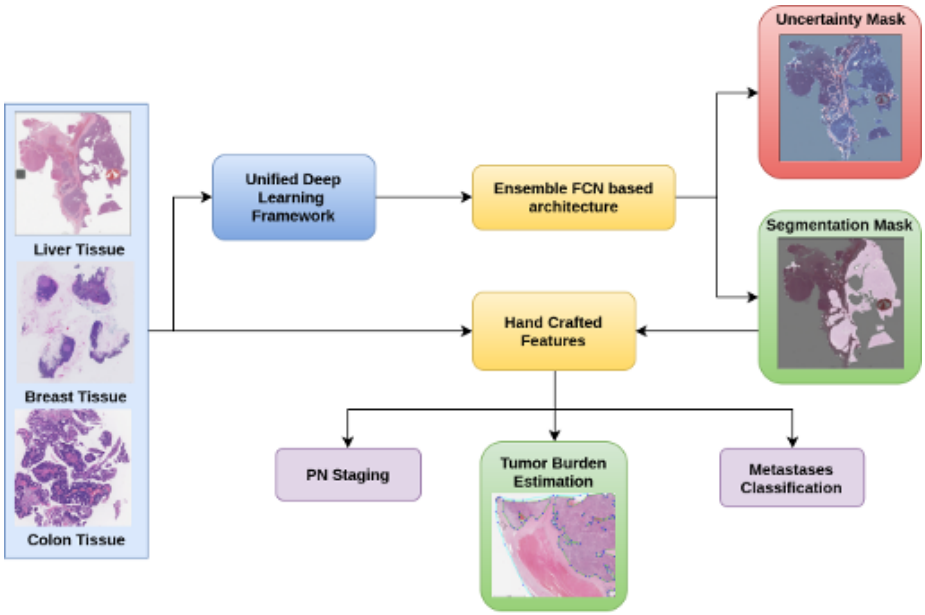


Figure 1: Entire framework pipeline

2.2.1. CAMELYON (CAncer MEtastases in LYmph nOdes challeNge)

The CAMELYON dataset comprises of sentinel lymph node tissue sections collected from the CAMELYON16 (CM16) and CAMELYON17 (CM17) challenges. The CM16 dataset comprised of 399 WSIs taken from two medical centers in the Netherlands, out of which 159 WSIs were metastases, and the remaining 240 were negative. Pathologists exhaustively annotated all the WSIs with metastases at the pixel level. In the CM16 challenge the 399 WSIs were split into training and testing sets, comprising of 160 negative and 110 metastases WSIs for training, 80 negative and 49 metastases WSIs for testing.

The CM17 dataset comprised of 1000 WSIs taken from five medical centers in the Netherlands. In CM17 challenge, 500 WSIs were allocated for training, and the remaining 500 WSIs were allocated for testing. The training dataset of CM17 included 318 negative WSIs and 182 WSIs with metastases. In CM17 dataset, slide-level labels were provided for all the WSIs and exhaustive pixel level annotations were provided for 50 WSIs. The slide-level labels were nega-

Table 3: Metastases type distribution in CAMELYON17 training set

Metastases (Training Set)			
Negative	ITC	Micro	Macro
318	35	64	88

tive, Isolated Tumor cells (ITC), micro-metastases and macro-metastases (Table 1). In CM17 challenge, the 1000 WSIs were divided into 200 artificial patients, and each artificial patient was constructed by grouping 5 WSIs from the same medical center. The pN-stage labels were provided for 100 patients from the training set based on the rules provided in Table 2. The Table 3 provides the metastases type distribution in CM17 training dataset.

2.2.2. DigestPath

DigestPath dataset consists of a total of 750 tissue slices taken from 450 different patients. On average, each tissue image was of size 5000x5000. The dataset also included fine pixel-level annotations of lesion done by experienced pathologists. Additionally, 250 tissue slices taken from 150 patients were provided as a test set. The data was collected from multiple medical centers, especially from several small centers in developing countries. All the tissues were stained by hematoxylin and eosin (H&E) and scanned at 20x resolution. Out of 750 tissue slices, 725 slices were used in training, and the rest 25 slices we considered as a held-out test set.

2.2.3. PAIP

The PAIP dataset contains a total of 100 whole slide images scanned from liver tissue samples. Each image had an average dimension of 50,000x50,000 pixels. All the images were stained by hematoxylin and eosin, scanned at 20x magnification, and prepared from a single center, Seoul National University Hospital. The dataset included pixel-level annotation of the viable tumor and whole tumor regions. It also provided the viable tumor burden metric for each image.

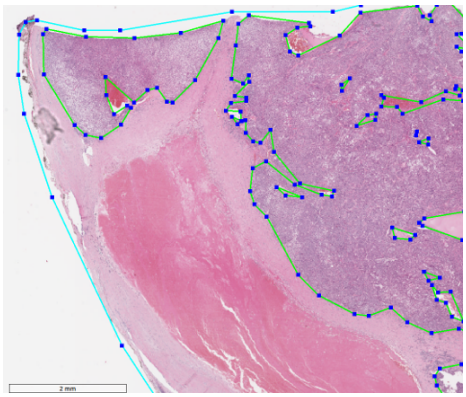


Figure 2: In the above figure green contour represents viable tumor and the blue contour represents whole tumor

Tumor burden is defined as the ratio of the viable tumor region to the whole tumor region. The viable tumor region is defined as the cancerous region. The whole tumor area is defined as the outermost boundary enclosing all the dispersed viable tumor cell nests, tumor necrosis, and tumor capsule (Fig 2). Each tissue sample contains only one whole tumor region. This metric has applications in assessing the response rates of patients to cancer treatment.

Out of the 100 images, 50 images were the publicly available training set, 10 images were reserved for validation set that was made publicly available after the challenge was completed, and the rest 40 images were the test set whose ground truth were not publicly available.

2.3. Data pre-processing

2.3.1. Tissue mask generation

Tissue mask generation was done to prevent unnecessary computations on blank spaces in tissue slice. In this step, the entire tissue region was separated from the background glass region of the slide. Here we make use of Otsu's adaptive thresholding (Otsu, 1979) technique on the low-resolution version of the WSI. The RGB color space of the low-resolution WSI was transformed to HSV (Hue-Saturation-Value) color space, and thresholding was applied to the saturation component. Post thresholding, morphological operations were done

Dataset	#WSIs	Average image size	#Number of patches extracted
CAMELYON	628	100,000x100,000	571,029
DigestPath	725	5,000x5,000	80,000
PAIP	50	50,000x50,000	200,000

to ensure that during patch-based inference of the WSI, the neighboring contexts were appropriately provided in the extracted patches at small tissue regions and tissue borders.

2.3.2. Patch coordinate extraction

The WSIs are gigapixel images, and the typical image sizes are of order $100k \times 100k$. We proposed to train our neural networks with small patches of fixed size extracted from the WSI. We extracted coordinates as the center of the patches from the low-resolution WSI's tissue mask. We rescaled the extracted coordinates to correspond to level-0 WSI. The randomly sampled coordinates from the WSIs corresponded to regions from the tumor, tumor boundary regions, and non-tumor tissue. From each WSI, a fixed number of patch coordinates corresponding to various regions of WSIs were extracted. Further, these extracted patch coordinates were split into three cross-validation folds.

2.3.3. Data augmentation, normalization and data loading

To increase the number of data points and generalization of models to various staining and acquisition conditions, we made use of data augmentations. Augmentations like flipping left, and right, 90-degree rotations, and Gaussian blurring along with color augmentation were performed. Color augmentation included random changes to brightness, contrast, hue, and saturation, the ranges for which were 64.0/255, 0.75, 0.25, 0.04, respectively. Additionally, in order to introduce a diversity of patches extracted from the WSI during every training cycle, we incorporated random coordinate perturbation. The random coordinate perturbation involved offsetting the center of the patch coordinates within

a specified radius prior to the extraction from WSI. We fixed our radius to be 128 pixels and extracted patches of size 256x256 from the level-0 image.

Post augmentation, the images were normalized using equation 1 to scale the range of data statistics, basically transforming the data to have zero mean and unit standard deviation. Since the image dimensions were huge, computing the image statistics during training was expensive. This was circumvented by assuming the mean and standard deviation of the image to be 128, which spreads the entire region.

$$X_{norm} = \frac{X - 128}{128} \tag{1}$$

The data-loader prepared batches of data comprising an equal number of tumorous and non-tumorous patches, thereby facilitating a balanced dataset for training the neural network.

2.4. Network architecture

For the task of segmentation of tumor regions from patches of WSIs, we propose to utilize encoder-decoder based fully convolutional neural (FCN) network-based architectures (Long et al., 2015). An encoder comprises of a series of operations (like convolution and pooling) that reduces the input size to a low dimension feature vector. The decoder upsamples the feature vectors to a larger dimension. The entire encoder-decoder is then trained end-to-end. Fully convolutional architecture substitutes fully connected layers with convolutional layers, thereby allowing input of arbitrary size. We propose an ensemble approach comprising of some of the FCN architectures that show the state of the art results on natural images, namely Densenet, InceptionResNetV2, and DeepLabV3Plus.

2.4.1. DenseNet

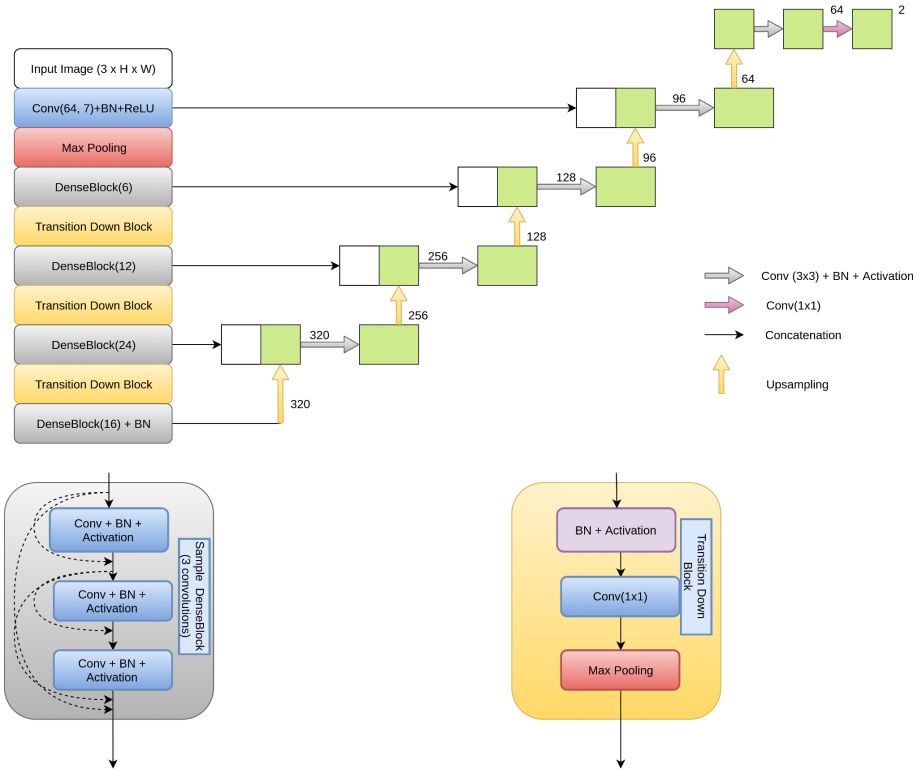


Figure 3: Densenet Architecture

We used a U-Net (Ronneberger et al., 2015) based encoder-decoder architecture. The main feature of U-Net is the presence of skip connections between layers of the encoder and decoder. These connections allow for the free flow of low-level information directly from the encoder to the decoder, bypassing the bottleneck. The skip-connection was made by concatenating the l^{th} layer with the $(n-l)^{th}$ layer.

For the encoder, we use the DenseNet (Iandola et al., 2014) architecture. The input for a particular layer in the DenseNet network was concatenated to inputs of all the previous layers. This was done to prevent the loss of information through forward pass in deep networks. The decoder was composed of the bilinear up-sampling module and convolutional layers.

2.4.2. *Inception-ResNetV2*

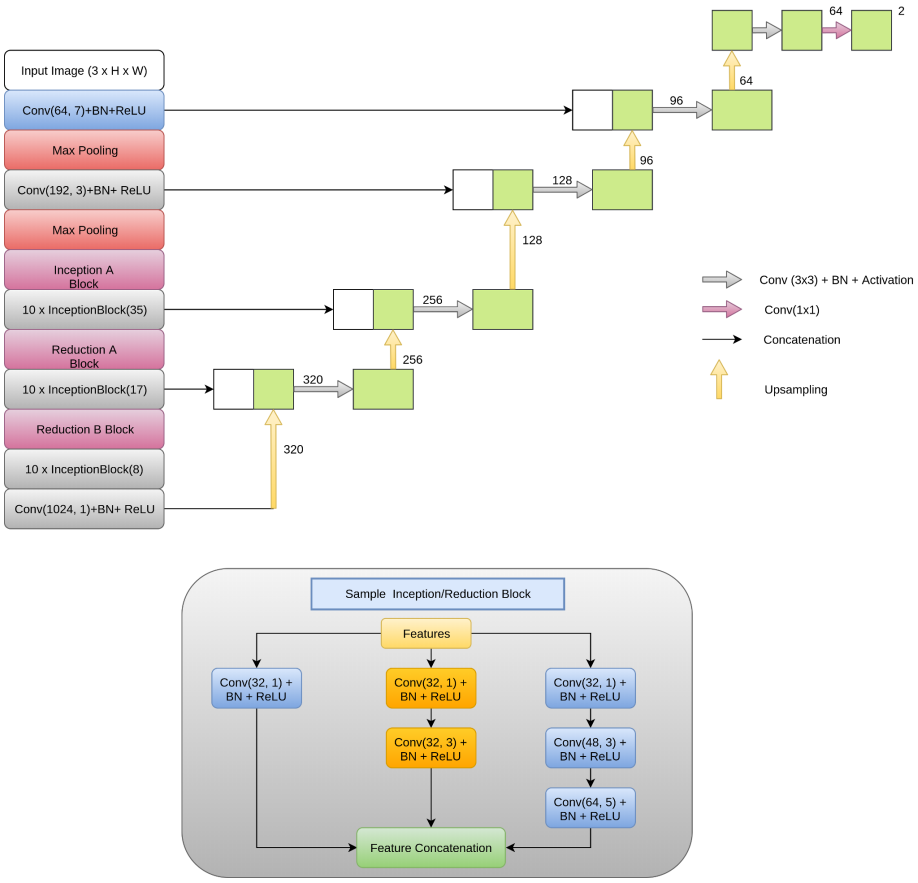


Figure 4: Inception-ResNetV2 architecture

The Inception-v4 (Szegedy et al., 2017) (also known as Inception-ResNetV2) integrates the features of the Inception (Szegedy et al., 2015) architecture and the ResNet (He et al., 2016) architecture. The ResNet architecture introduced skip-connections between layers by summing the input and output of the $(l-1)^{th}$ layer and feeding that as the input for the l^{th} layer. The Inception architecture contains inception blocks that have convolutions of various sizes. This waives the need to deliberate on the filter sizes for each layer.

In the Inception-ResNet's architecture, the inception blocks have skip con-

nections that sum the input of the block to the output.

2.4.3. DeepLabV3Plus

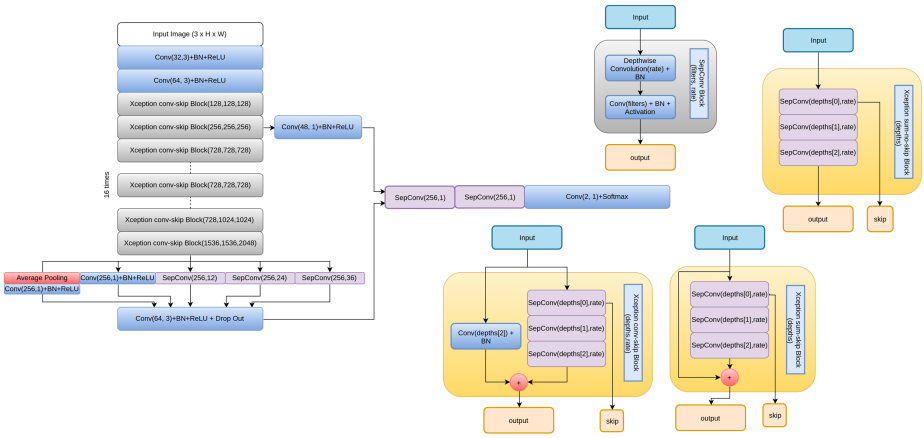


Figure 5: DeepLabV3Plus architecture

DeepLabV3 (Chen et al., 2017) was built to obtain multi-scale context. This was done by using atrous convolutions with different rates. DeeplabV3Plus (Chen et al., 2018) extends this by having low-level features transported from the encoder to decoder.

2.4.4. Ensemble

Our ensemble comprised of 3 FCN models, each of the individual FCN model was trained on one of the 3-fold cross-validation splits. During inference, the predicted posterior probabilities of the segmentation maps were averaged to generate the ensemble model prediction. The models were trained and inferred on the image patch size of 256x256 at the highest resolution of WSI.

2.5. Loss function

Tumor regions were represented by a minuscule proportion of pixels in WSIs, thereby leading to class imbalance. This issue was circumvented by training the network to minimize a hybrid loss function. The hybrid cost function comprised of cross-entropy loss and a loss function based on the Dice overlap coefficient.

The dice coefficient is an overlap metric used for assessing the quality of segmentation maps. The dice loss is an differentiable function that approximates to dice-coefficient and is defined using the predicted posterior probability map and ground truth binary image as defined in the Eq. 2. The cross-entropy loss is defined in Eq. 3. In the equations, p_i is the predicted posterior probability map, and g_i is the ground truth image.

$$DiceLoss = 1 - \frac{2\sum_i^N p_i g_i}{\sum_i^N p_i^2 + \sum_i^N g_i^2} \tag{2}$$

$$CrossEntropyLoss = \sum_i^N (g_i log(p_i) + (1 - g_i) log(1 - p_i)) \tag{3}$$

The total loss is defined as a linear combination of the two loss components as defined in Eq. 4. The neural networks are trained by minimizing the total loss. The α, β, γ are empirically assigned to the individual loss components. FG and BG represent the foreground pixels that correspond to the tumor regions and the background pixels that corresponded to non-tumor regions, respectively. In this work we set $\alpha = 0.5, \beta = 0.25$ and $\gamma = 0.25$.

$$TotalLoss = \alpha * CrossEntropyLoss + \beta * DiceLossBG + \gamma * DiceLossFG \tag{4}$$

2.6. Uncertainty analysis

Uncertainty estimation is essential in assessing unclear diagnostic cases predicted by deep learning models. It helps pathologists/doctors to concentrate more on the uncertain regions for their analysis. The need and challenges that we might face in the context of uncertainty are discussed in (Begoli et al., 2019). There exist two main sources of uncertainty, namely (1) Aleatoric uncertainty, and (2) Epistemic uncertainty. Aleatoric uncertainty is uncertainty due to the data generation process itself. In contrast, the uncertainty induced due to the model parameters, which is the result of not estimating ideal model architectures or weights to fit the given data, is known as epistemic uncertainty (Kendall and Gal, 2017). Epistemic uncertainty can be approximated by using test time

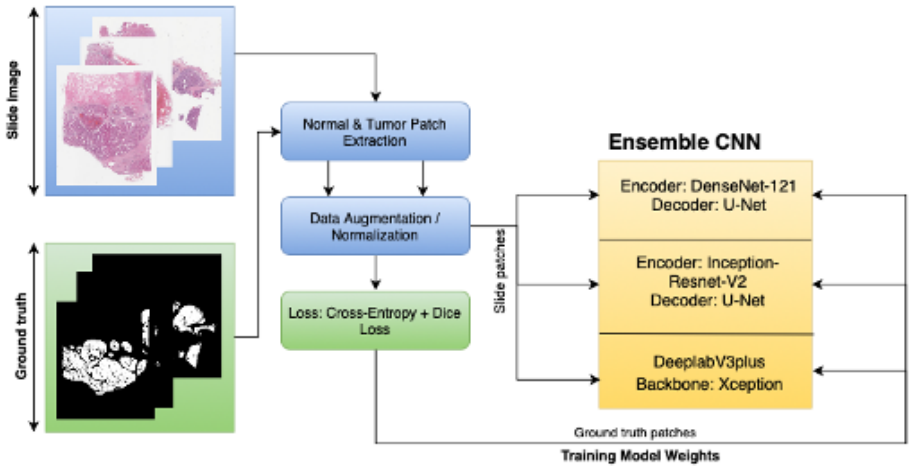


Figure 6: Overview of training pipeline

Bayesian dropouts as discussed in (Leibig et al., 2017), which estimates uncertainty by Montecarlo simulations with Bayesian dropout.

In the proposed pipeline, we estimate aleatoric uncertainty for each model using test time augmentations, as introduced in (Gal and Ghahramani, 2016). For epistemic uncertainty, we make use of the diversity of model architectures and calculate uncertainty as described in equation 5.

$$var_{epistemic}(x) \approx \mathbf{E}_{\phi \sim \{\Phi_i\}}(\phi(x|w) - \mathbf{E}_{(\phi \sim \{\Phi_i\}}(\phi(x|w)))^2) \tag{5a}$$

$$var_{aleatoric}(x, \Phi_i) \approx \mathbf{E}_{t \sim TTA}(\Phi_i(x|w, t) \tag{5b}$$

where Φ_i indicates trained model, where $\Phi_i \in \{\Phi_{densenet}, \Phi_{inception}, \Phi_{deeplabv3}\}$ and TTA denotes the set of possible test time data augmentations allowed. In our case $TTA \in \{rotation, vertical flip, horizontal flip\}$.

2.7. Training pipeline

Figure 6 describes the training strategy utilized for training each of our models. The batches for training were generated with an equal number of tumor and non-tumor patches. This was done to prevent class imbalance or

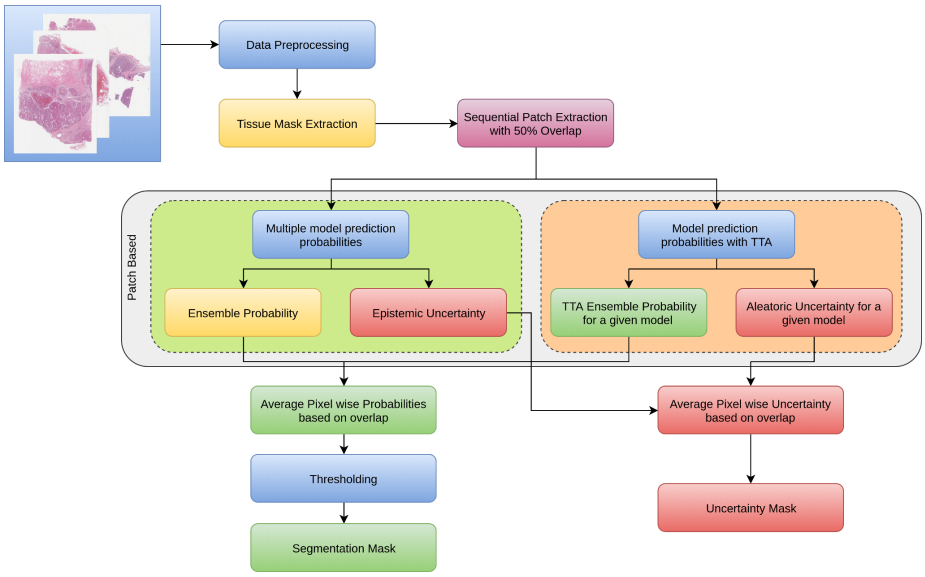


Figure 7: Overview of Inference Pipeline

manifold shift issues and enforce proper training. All three models were trained independently, with different folds of the data.

We made use of the transfer learning technique by using ImageNet(Deng et al., 2009) pre-trained weights for encoders based on DenseNet and Inception-ResNetV2. We made use of the models pre-trained on PascalVOC(Everingham et al., 2010) in the case of DeeplabV3Plus. For the first two epochs, the encoder section of the models were frozen, and only the decoder weights were made trainable. For the remaining epochs, both the encoder and decoder parts were trained, and the learning rate was reduced gradually. The learning rate was decayed after every few epochs in a deterministic manner to allow for the model to gradually converge.

2.8. Inference pipeline

Figure 7 provides an overview of the inference pipeline to generate tumor probability maps and model uncertainty maps. In a typical WSI, the tissue region occupies a smaller fraction when compared to the background glass slide, hence for faster inference, the pre-processing step involved segmentation of tis-

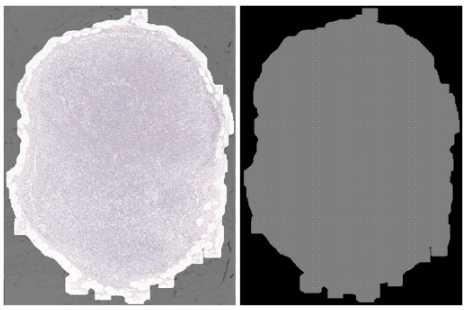


Figure 8: (Left to Right) A tissue mask overlayed on the WSI image at low resolution (level-4), here the white region corresponds to the tissue mask; Uniform Patch Coordinate Sampling Grid, here the points on the image act as centers from which high-resolution image patches were extracted from the WSI.

sue region. This tissue segmentation mask was generated, as discussed in section 2.3.1. In order to facilitate extraction of patches from the WSI within the tissue mask region a uniform patch-coordinate sampling grid was generated at lower resolution as shown in Figure 8. Each point in the patch sampling grid was rescaled by a factor to map to the coordinate space corresponding to WSI at its highest resolution. From these scaled coordinate points as the center, fixed-size high-resolution image patches were extracted from the WSI. We define sampling stride as the spacing between consecutive points in the patch sampling grid. The high-resolution patch size and the sampling stride controlled the overlap between consecutive extracted patches from the WSI. The main drawback with patch-based segmentation of WSI was that the smaller patches could not capture a wider context of the neighborhood. Moreover, stitching of the patch segmentation introduced boundary artifacts (blockish appearance) in the tumor probability heat-maps. We observed that the generated heat-maps were smooth when the inference was done on overlapping patches with larger patch-size and averaging the prediction probabilities at overlapping regions. Our experiments suggested that 50% overlap between consecutive neighbouring patches was the optimal choice as it ensured that a particular pixel in the WSI was seen at most 4-times during the heat-map generation. However, this approach increased the

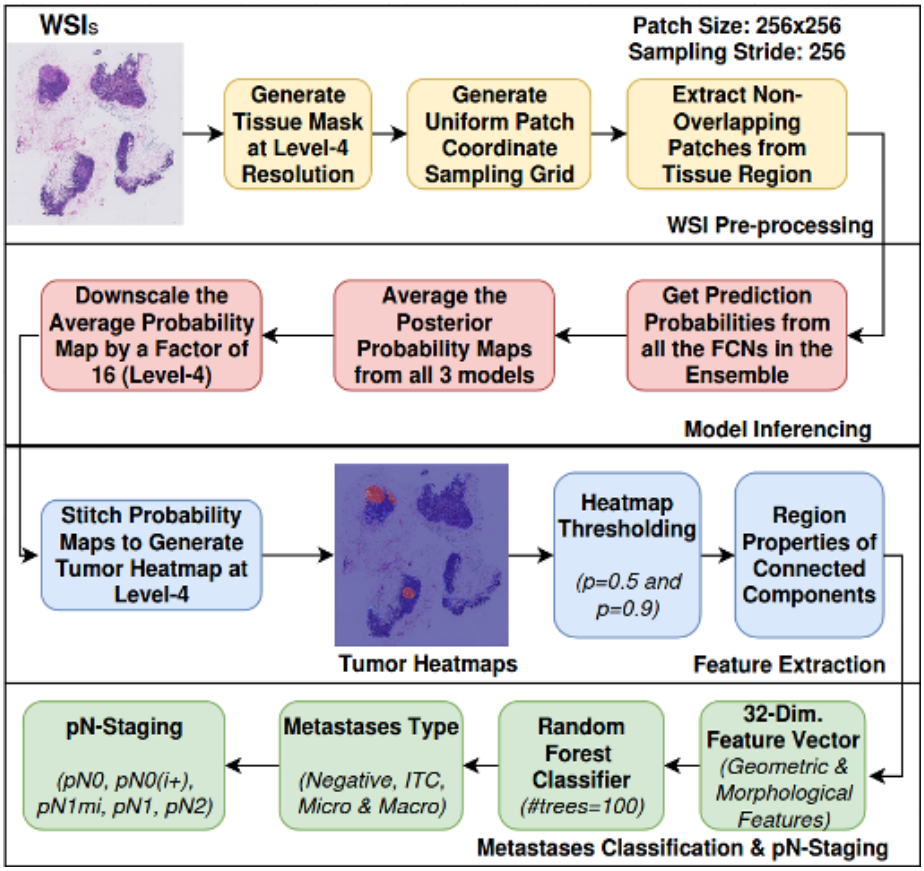


Figure 9: Overview of pN-Staging pipeline

inference time by a factor of 4. We also observed that during inference increasing the patch size by a factor of 4 when compared to the patch size used during training (256x256) improved the quality of generated heat-maps. The generated heat maps obtained by single model with multiple input images varied with different test time augmentations were used in the estimation of aleatoric uncertainty map. Epistemic uncertainty map was estimated by using multiple heat maps obtained from different models for a given input image.

Table 4: Features extracted from the heatmap for predicting lymph node metastases type.

No.	Feature description	Threshold (p)
1	Largest tumor regions major axis length	p=0.9 & p=0.5
2	Largest tumor region's area	p=0.5
3	Ratio of tumor region to tissue region	p=0.9
4	Count of non-zero pixels	p=0.9
5	Tumor regions area: maximum, mean, variance, skewness, and kurtosis	p=0.9
6	Tumor regions perimeter: maximum, mean, variance, skewness, and kurtosis	p=0.9
7	Tumor regions eccentricity: maximum, mean, variance, skewness, and kurtosis	p=0.9
8	Tumor regions extent: maximum, mean, variance, skewness, and kurtosis	p=0.9
9	Tumor regions solidity: maximum, mean, variance, skewness, and kurtosis	p=0.9
10	Mean of all region's mean confidence probability	p=0.9
11	Number of connected regions	p=0.9

2.9. pN-Staging Pipeline

The Figure 9 illustrates the overall pipeline for pN-Staging. The pipeline comprises 4 blocks as described below:

- Pre-processing: The tissue regions in the WSIs were detected for patch extraction.
- Heatmap generation: The extracted patches from the WSIs were passed through the inference pipeline to generate the down-scaled version of the tumor probability heatmaps.
- Feature extraction: The heatmaps were binarized by thresholding at 0.5 and 0.9 probabilities, and at each of these thresholds, the connected components were extracted, and region properties were measured using scikit-image(van der Walt et al., 2014) library. We computed 32 geometric and morphological features of the probable metastases regions. Table 4 provides a description of all the features computed.
- Classification: The pN-stage was assigned to the patient based on all the available lymph-node WSIs, taking into account their individual metastases type (Table 1). We employed a simple count-based rule based on

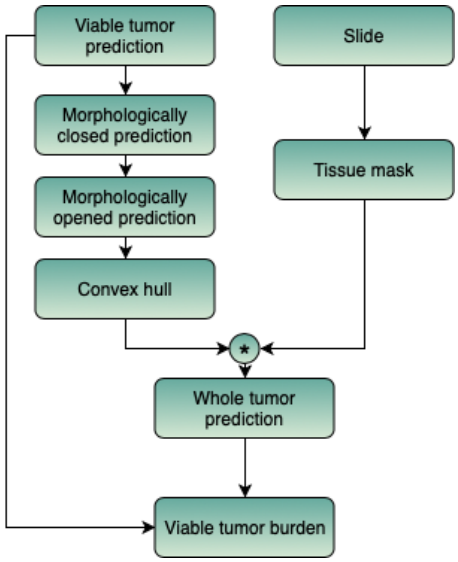


Figure 10: Overview of viable tumor burden pipeline

pN-staging as defined in Table 2. For predicting the metastases type, we built an ensemble of Random Forest classifiers (Liaw et al., 2002) using the extracted features.

2.10. Tumor burden estimation

We obtained the viable tumor region from the segmentation pipeline discussed. For the whole tumor region, initially, we attempted to use the same segmentation pipeline and model it as a binary classification problem. However, the model provided results that had an accuracy lesser than 10%. Therefore, we adopted a heuristic method to calculate the tumor burden. First, the viable tumor region was predicted. Morphological operations were applied to remove some of the false positives and fill the holes. Next, the smallest convex hull that contained the entire viable tumor region was calculated. Simultaneously, a tissue mask was obtained using otsu thresholding. The convex hull was then multiplied with the tissue mask to obtain the whole tumor segmentation. The viable tumor burden was then calculated by taking the ratio of the area of the

viable tumor and whole tumor. Figure 10 show an overview of the viable tumor burden estimation. The results are shown in Figure 18.

3. Experimental analysis

In this section, we experimentally analyze the effectiveness of our proposed methodologies for segmentation and classification models. We implemented our neural networks using TensorFlow (Abadi et al. (2016)) software. We ran our experiments on multiple desktop computers with NVIDIA Titan-V GPU with 12 GB RAM, Intel Core i7-4930K 12-core CPUs @ 3.40GHz, and 48GB RAM.

3.1. Lesion detection performance on CAMELYON

In this section, we detail some of the techniques specific to CAMELYON dataset pre-processing and discuss the performance of various FCN architectures and ensemble configurations for lesion detection on the CAMELYON16 test dataset (n=139).

3.1.1. Tissue mask generation

In some of the CM17 cases, the Otsu's thresholding failed because of the black regions in WSI. Hence, prior to the application of image thresholding operation, the pre-processing involved replacing black pixel regions in the WSI back-ground with white pixels and median blurring with a kernel of size 7x7 on the entire image. Median blur aided in the smoothing of the tissue regions and removal of noise at the tissue bordering the glass-slide region‘ while preserving the edges of the tissue. Figure 11 illustrates the pipeline for tissue mask generation with an example.

3.1.2. Dataset preparation

For training our model for lesion segmentation, we used training sets of both CAMELYON16 and CAMELYON17 dataset, which had pixel-level annotations. As noted by the challenge organizers, some of the WSIs were not exhaustively annotated in the CAMELYON16 training set; we excluded them in our training

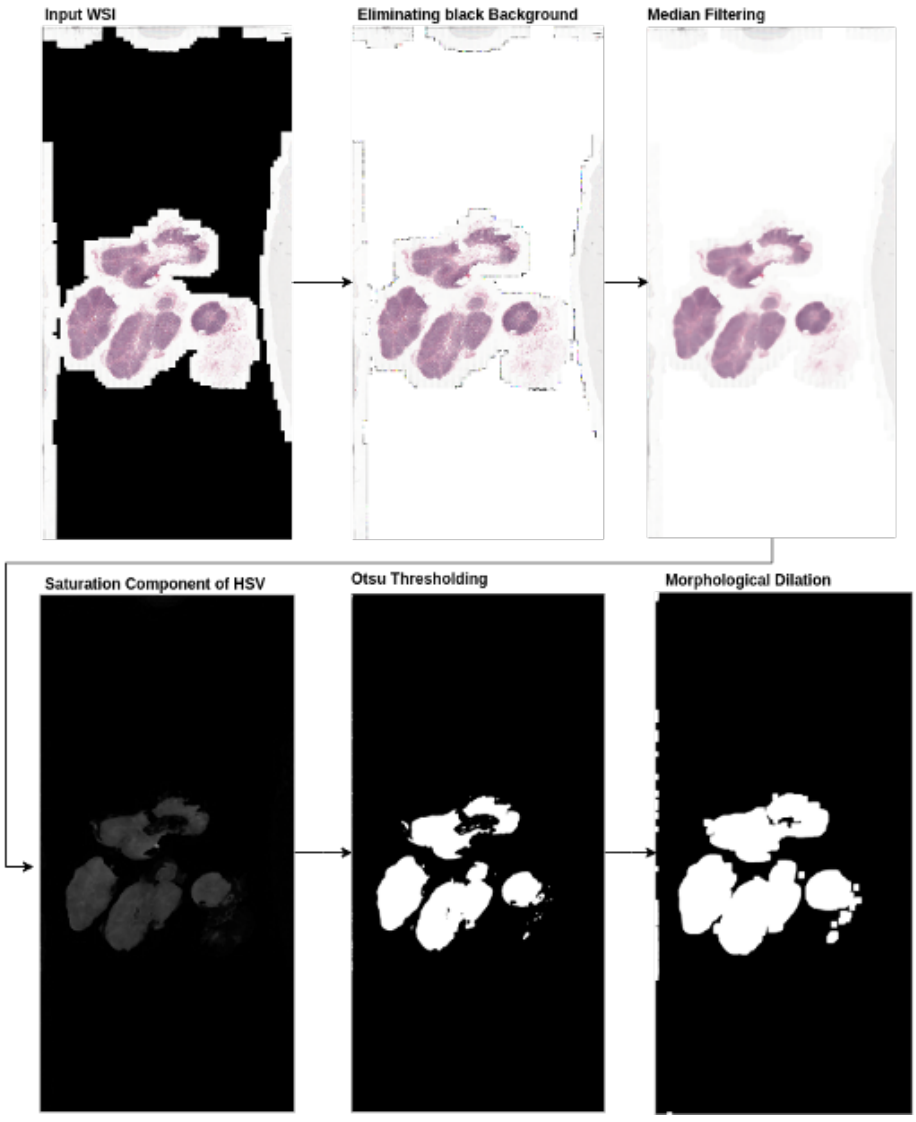


Figure 11: An illustration of the intermediate results of tissue mask generation of WSI in CM17 dataset.

Table 5: The table gives the statistics of the tumor and non-tumor patches in each of the cross-validation folds.

Patch label	**No. of patch coordinates**		
	Fold-0	**Fold-1**	**Fold-2**
Training Non-Tumor	187034	196094	190424
Training Tumor	184467	194709	187440
Validation Non-Tumor	99742	90682	96352
Validation Tumor	99786	89544	96813

set preparation. So, in total, we had 628 WSIs for training (250 WSIs from CAMELYON16 and 378 from CAMELYON17). We made a 3-fold stratified cross-validation split of the training set to maximize the utilization of the limited number of WSIs. The stratification ensured that the ratio of negative to metastases was maintained in all the three folds. From 628 WSIs, we randomly sampled 571029 coordinates whose patches included regions from the tumor and non-tumor tissue regions. A patch extracted from a WSI was labeled as a tumor patch if it had non-zero pixels labeled as tumor pixels in the pathologists manual annotation. Further, these extracted patch coordinates were split into 3-cross validation folds. Table 5 shows the distribution of the split in each of the folds.

3.1.3. Training and inference configuration of ensemble FCN models

We experimented with following two types of Ensemble configuration:

- Ensemble-A: It comprised of the three different FCN architectures, as described in section 2.7. The inference pipeline made use of patch size of 256 and extracted non-overlapping patches, as illustrated in Figure 9.
- Ensemble-B: It comprised of three replicated versions of a single FCN architecture. The inference pipeline made use of a patch size of 1024 with a 50% overlap between neighboring patches, as illustrated in Figure 12.

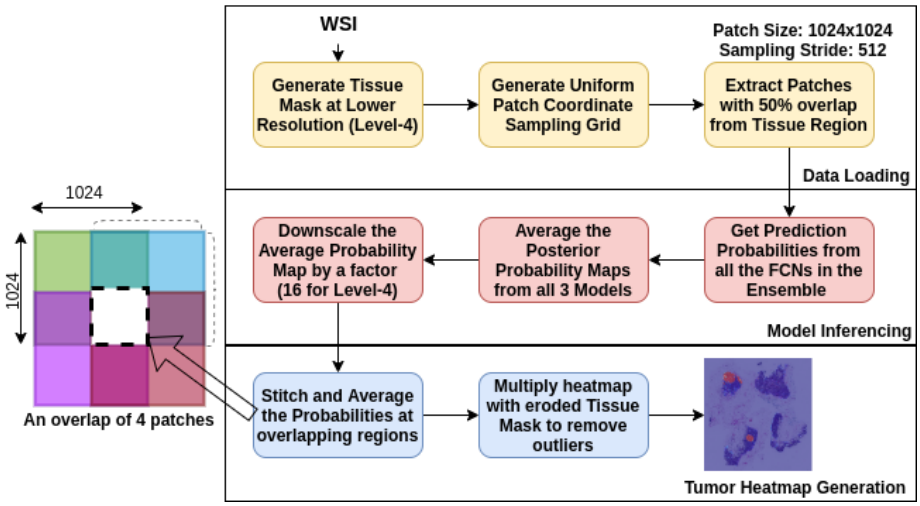


Figure 12: The figure illustrates the overlap-stitch inference pipeline used in Ensemble-B configuration.

In both the ensemble configurations, each model in the ensemble was trained with one of the 3-fold cross-validation splits. All the models made use of pre-trained weights with the fine-tuning procedure, as described in section 2.7. The models were trained for ten epochs with a batch size of 32.

3.1.4. Lesion detection performance of Ensemble-A

Table 6 provides a summary of the Ensemble-A and its individual constituting FCN models. Our results showed that DenseNet-121 architecture had higher sensitivity and reduced False positives when compared to other FCNs in the ensemble configuration. It was also observed that Ensemble-A showed significant difference in FROC score (Appendix A.0.3) compared to its constituents. The reason for this significant boost in the performance of Ensemble-A could be attributed to the effect of averaging the heatmaps from multiple FCN models, thereby lowering the probabilities of uncertain or less confident regions and hence eliminating the False Positives. Figure 13 illustrates the heatmaps generated by Ensemble-A and its constituent FCN models on a CAMELYON16 test case.

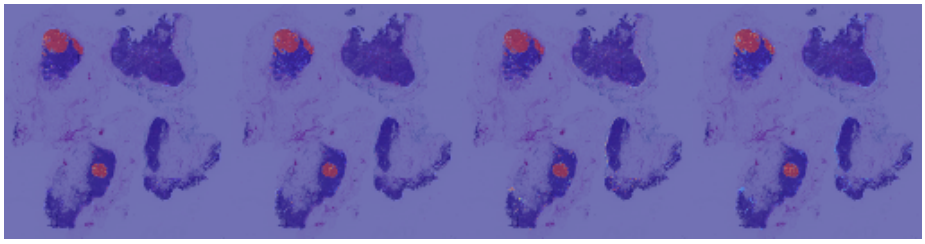


Figure 13: The figure shows the heat-maps overlayed on the WSI by FCN models in Ensemble-A configuration. (Left to Right): DenseNet121 FCN, InceptionResNetV2 FCN, DeepLabV3plus, Ensemble, (Patch Size: 256x256, Sampling Stride: 256 pixels).

Table 6: The table shows the FROC scores achieved on CAMELYON16 test set (n=139) by FCN models in Ensemble-A configuration. Note the abbreviations: IRF - Inception-ResNetV2 FCN, DF- DenseNet-121 FCN, DL- DeepLabV3Plus, FP- False Positives

Average FPs /WSI	**Sensitivity**			
	IRF (Fold-0)	**DF (Fold-1)**	**DL (Fold-2)**	**Ensemble-A**
0.25	0.5	0.59	0.03	0.77
0.5	0.59	0.65	0.06	0.8
1	0.69	0.72	0.2	0.83
2	0.77	0.79	0.48	0.84
4	0.8	0.83	0.64	0.86
8	0.82	0.85	0.77	0.86
FROC Score	0.69	0.74	0.36	**0.83**

3.1.5. Lesion detection performance of Ensemble-B

From Table 6 it was evident that the performance of DenseNet-121 FCN performed significantly better than the other two models in Ensemble-A. Hence, we proposed to design an Ensemble-B comprising of three DenseNet-121 FCNs as its constituents, each model was trained on one of the 3 cross-validation folds. The results are summarized in Table 8. We observed an improvement in FROC score for Ensemble-B (Appendix A.0.3) when compared to Ensemble-A. We also observed that the performance of individual models in the ensemble were similar (considering FROC score). However, this observation contrasted with Ensemble-A, where the models showed significant difference in their FROC scores. This also helped us to ascertain the fact that the performance difference in individual FCN models of Ensemble-A were not because of data variation seen between individual cross-validation folds. The Figure 14 illustrates the heatmaps generated by Ensemble-B and its constituent FCN models on a CAMELYON16 testcase. The Table 7 provides a summary of FROC scores on CAMELYON16 testset (n=139) for various configurations of patch-size and sampling stride. We observed that running inference on larger patch size enabled generation of smoother heatmaps and patch extraction with overlapping stride gave better context at the patch borders thereby minimizing the blockish artifacts seen in the heatmaps. In the Figure 14 we observed that while running inference on larger patch size, the model slightly struggled at tissue borders which was evident from the confidence probability values. This was primarily because we had trained the models on patch size of 256 and our training samples lacked regions from tissue border and glass-slide regions. Since, there was a marginal difference in FROC scores between various ensemble configurations (Table 7), in the interest of minimizing the computation time we preferred to use Ensemble-A with patch-size and sampling stride set to 256 (non-overlapping patch acquisition) for running inference on CAMELYON17 testing dataset (n=500).

Table 7: The table shows the FROC scores on CAMELYON16 test set (n=139) for various configurations of model, patch-size, and sampling-stride.

Model	**Patch size**	**Sampling stride**	**FROC score**
Ensemble-A	256	256	0.83
Ensemble-B	256	256	0.84
Ensemble-B	1024	1024	0.86
Ensemble-B	1024	512	0.85

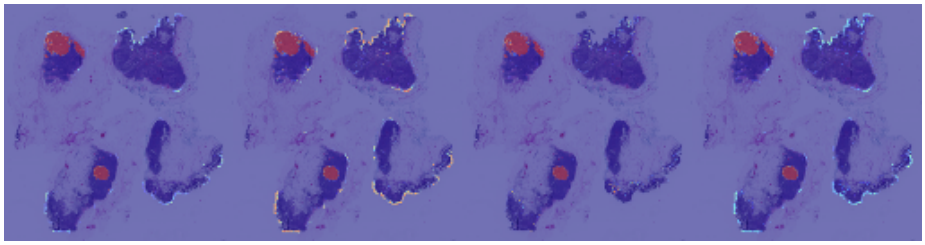


Figure 14: The figure shows the heat-map overlayed on the WSI by FCN models in Ensemble-B configuration. (Left to Right): 3 DenseNet121 FCN models trained on cross-validation folds: 1, 0 and 2 respectively and Ensemble-B, (Patch Size: 1024x1024, Sampling Stride: 512 pixels).

Table 8: The table shows the FROC scores achieved on CAMELYON16 test set (n=139) by FCN models in Ensemble-B configuration. Note the abbreviations: DF- DenseNet-121 FCN, FP- False Positives.

Average FPs /WSI	**Sensitivity**			
	DF (Fold-0)	**DF (Fold-1)**	**DF (Fold-2)**	**Ensemble-B**
0.25	0.56	0.56	0.61	0.77
0.5	0.65	0.63	0.69	0.84
1	0.71	0.70	0.76	0.85
2	0.77	0.75	0.81	0.88
4	0.82	0.80	0.84	0.88
8	0.86	0.86	0.88	0.89
FROC Score	0.73	0.72	0.77	**0.85**

Table 9: Details of our CM17 train and validation set split

Dataset	**No. of WSIs per each metastasis type**				
	Negative	ITC	Micro	Macro	Total
Train set	100	26	35	44	215
Validation set	318	35	64	88	285

3.2. Lymph-node metastases type classification

In this section we detail and discuss the dataset preparation and training methodology of Random Forest classifiers for metastases type classification.

3.2.1. Dataset preparation

The CM17 training dataset (Table 3) had 100 patients and each patient had 5 WSIs with their corresponding metastases labels (total 500 WSIs). We split 100 training patients into 43 patients as train set and the remaining 57 patients as validation set. The train set comprised of patients which had atleast one of their WSIs with pixel level annotation. The Table 9 shows the distribution of WSIs in terms of metastases type between train and validation sets, our split strategy ensured that the distribution of metastases type between the two splits were similar.

3.2.2. Training methodology

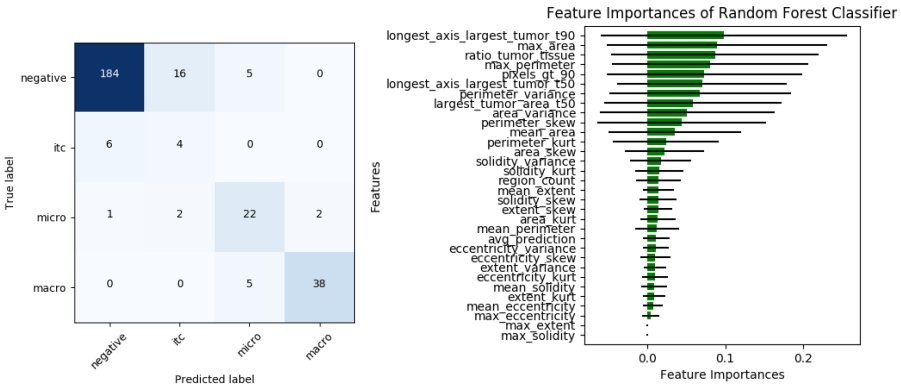
We generated tumor probability heatmaps for all the 500 WSIs using Ensemble-A configuration (section 3.1.3) and extracted all the features listed in Table 4. Post generation of features, we cleaned the training set by removing some of the outlier points. The outliers were detected based on threshold-based heuristics like the presence of significantly large tumor false-positive regions in negative cases, etc. For the purpose of classifier selection, feature elimination and hyper-parameter tuning we initially trained our classifiers on the train set (n=215) and validated on the held-out validation set (n=285). Our experimentation on various classifiers showed that the optimal performance in-terms of classification

accuracy (90.18%) and Cohen's kappa score (0.9164) on held-out validation set was achieved with Random Forest classifier with 100 trees. Figure 15 shows the confusion matrix and the feature importance ranking by the Random Forest classifier. From Table 9, we observed that the data distribution was highly class imbalanced, with negative cases being majority class and ITC cases being minority class. This lead to mis-classifications between ITC and negative cases, as evident in the confusion matrix.

We further experimented by training another Random Forest classifier on the complete training set (n=500) in order to maximize the utilization of training points. The 5-fold cross-validation showed an average accuracy score of 90%, and its performance was similar to the model trained on train set (n=215). We refer to the above two trained models as RF-PI and RF-CI (Random Forest classifiers trained on the partial and complete training set with imbalanced class data, respectively).

3.2.3. Data balancing

In order to handle the class imbalance problem, one of the techniques proposed in the literature is oversampling by synthetically generating minority class samples using SMOTE algorithm. (Chawla et al., 2002). However, this method can introduce noisy samples when the interpolated new samples lie between marginal outliers and inliers. This problem is usually addressed by removing noisy samples by using under-sampling techniques like Tomek's link (Tomek, 1976) or nearest-neighbors. We proposed to balance our training data using a SMOTETomek (Batista et al., 2004) algorithm, a combination of SMOTE and Tomek's link performed consecutively. We separately balanced our train set (n=215) split and the complete training set (n=500). We independently trained two Random Forest classifiers using these two balanced datasets. We refer to these two trained models as RF-PB and RF-CB (Random Forest classifiers trained on Partial and Complete training set, which are class Balanced data respectively). Table 10 provides the results of the validation study performed on all four models. We observed that post-class balancing of the training dataset,



Confusion Matrix of RF-PI

Feature Importance

Figure 15: (a) Confusion Matrix of Random Forest classifier trained on imbalanced train set predicted on held-out validation set, (b) Feature importance ranking by Random Forest classifier for metastases classification. The green bars are the feature importance of the forest, along with their inter-trees variability (standard deviation). The feature importance shows that the most important feature for metastases classification is largest tumor region's long axis/diameter, which corroborates to pathologist way of determining the metastases type from the WSI. Note: _t90 and _t50 indicate heatmaps thresholded at 0.9 and 0.5 probability values, pixels_gt_90 indicate number of non-zero pixels in the binary image after thresholding the heatmap 0.9 probability.

Table 10: The table provides the validation results of the four Random Forest classifiers, each trained on different subsets of the training data. Note: For the models RF-PI and RF-PB, held-out validation existed, whereas, for the other two models, it was not available as it was trained on the entire training set, and hence N.A (not applicable) is mentioned in the table. For all the models, 5-fold cross-validation accuracy was estimated on their respective training sets. These values are provided as mean (standard deviation).

	Accuracy (%)	
Classifier	**5-fold CV**	**Validation set**
RF-PI	87 (0.06)	90.18
RF-PB	92 (0.03)	87.02
RF-CI	89.89 (0.03)	N.A
RF-CB	94.83 (0.02)	N.A

Table 11: Segmentation results on held-out test set of DigestPath

Models	Dice
DeepLabV3Plus	0.81
DenseNet-121 FCN	0.84
Inception-ResNetV2 FCN	0.84
Ensemble	0.86

the 5-fold cross-validation accuracy scores improved by a margin of 5%.

3.2.4. Ensembling classifiers

We proposed an ensemble strategy for combining the predictions of all the four trained Random Forest classifiers. The ensembling was based on the majority voting principle, and in case of a tie, the higher metastases category was selected. We refer to this model as RF-Ensemble.

3.3. Segmentation performance on DigestPath

In this section we discuss specific details on the training and inference strategies on DigestPath dataset. We split the training data (n=725) into 3-fold cross-

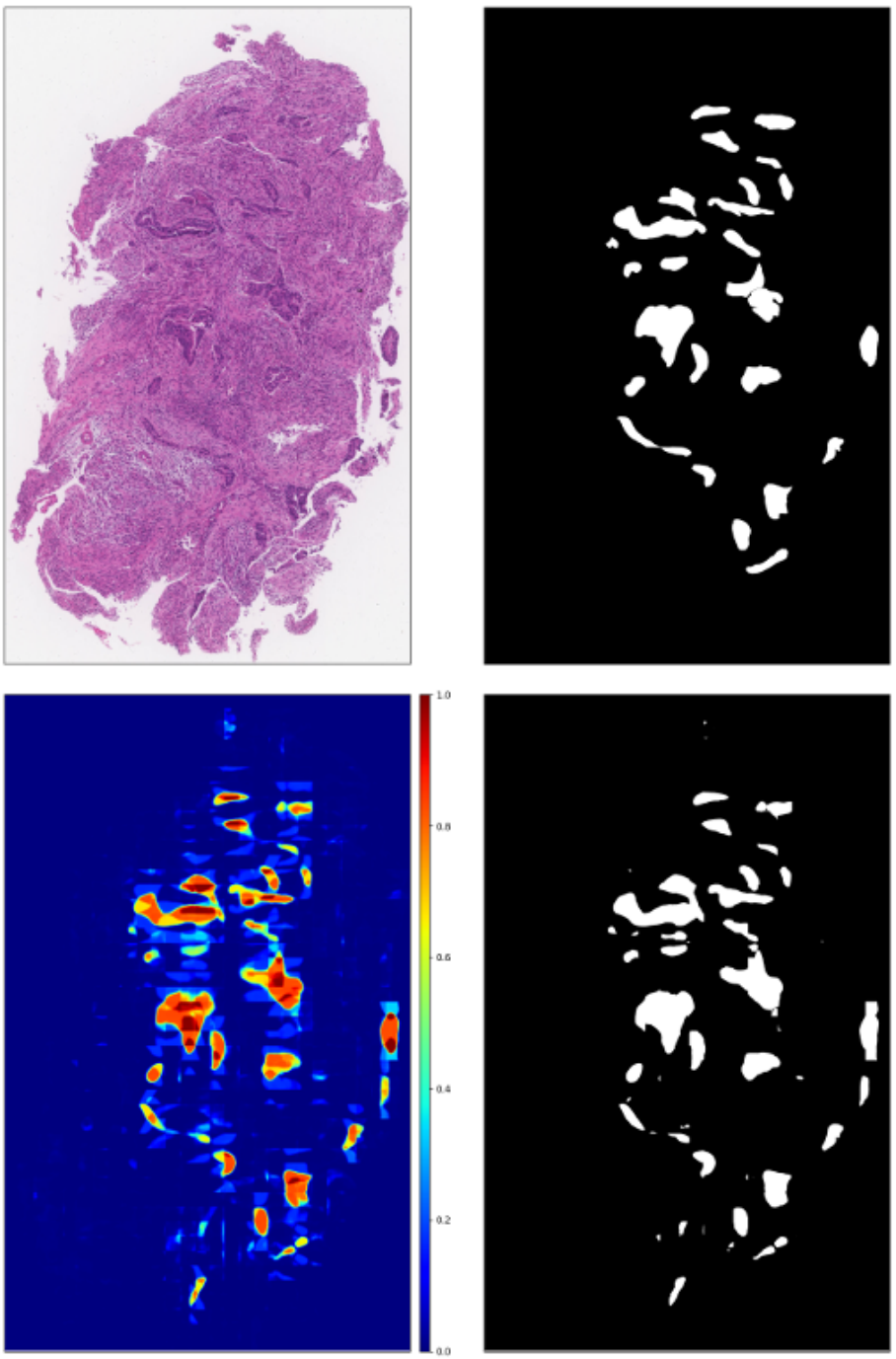


Figure 16: (Clockwise from top left) WSI Image; Ground truth of the tumor; Prediction probability of the ensemble pipeline; Prediction with threshold of 0.5

validation sets, each set of data was used to train the individual models in the ensemble. We extracted a total of 80,000 patches from the entire training data. We trained each model with a patch size of 256 and a batch size of 32, with Adam optimizer.

In the case of inference, we made use of patches with dimension 256 x 256, with 50% of overlap and a batch size of 32. We considered 0.5 as a threshold value to generate a segmentation binary mask on a predicted probability map.

Figure 16 illustrates an example of the segmentation map generated by our ensemble model. We tested our trained models on a subset of the training set (n=25) reserved for our internal testing set, and their corresponding results are tabulated in Table 11.

3.4. Segmentation performance on PAIP

In this section we elaborate on the training and inference details for segmentation on the PAIP dataset. For the tissue mask, we first performed Otsu thresholding on the HSV version of the slide image. We then performed the closing morphological operation with a kernel size of 20, followed by an opening operation with a kernel size of 5 and finally a dilation operation with a kernel size of 20.

We split the training data (n=50) into five cross-validation folds and used three of those folds for training the models of the ensemble. The data was split into five folds as opposed to three folds to ensure that each training set had at least 40 samples. We extracted a total of 200,000 patches from the entire training data with equal contributions from each training sample. We trained with a patch size of 256 and a batch size of 32.

For the inference, we used patches of 1024 and a batch size of 16. For the threshold, we experimented with a range of values and chose 0.5 as it gave optimal segmentation performance on the validation set(n=10). We observed that setting lower threshold(0.5) resulted in false positives and higher threshold(0.9) led to under-segmentation.

Figure 17 illustrates an example of the segmentation map generated by our

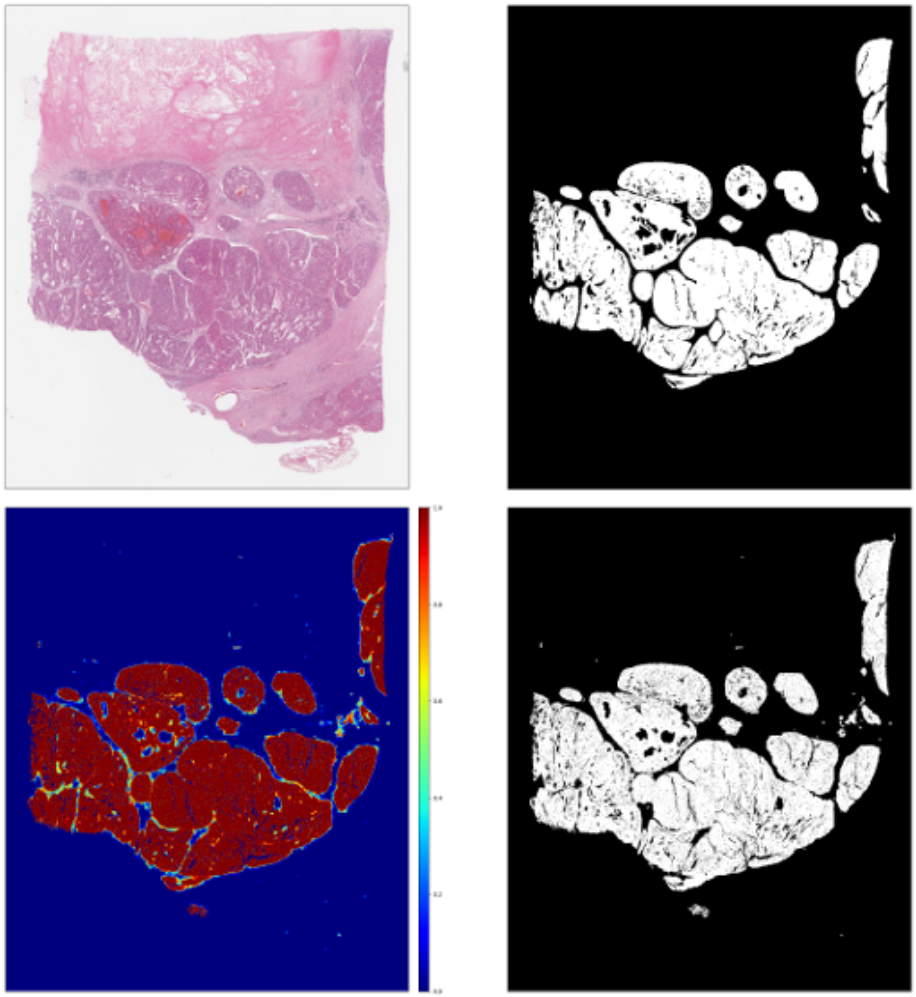


Figure 17: (Clockwise from top left) WSI Image; Ground truth of the tumor; Prediction probability of the ensemble pipeline; Prediction with threshold of 0.5

Table 12: Segmentation results on validation set of PAIP

Model	Jaccard Score
DeepLabV3Plus	0.681786
Inception-ResnetV2	0.685771
Densenet	0.679107
Ensemble	0.701621

ensemble model. We tested our trained models on the validation set (n=10), and their corresponding results are tabulated in Table 12.

3.5. Viable Tumor Burden

Figure 18 shows the whole tumor region predictions obtained by the methodology described 2.10. The first image shows that this pipeline gives a good approximation to the whole tumor region. Most of the samples fit into this category. Our pipeline failed in two scenarios. One is when the segmentation pipeline predicted viable tumors in discrete dispersed regions. This is illustrated in the second example in Figure 18 where there is a spurious prediction apart from the main tumor prediction. The other failure case is when the actual whole tumor region is larger than the convex hull boundary of the actual viable tumor region. The third sample in figure 18 demonstrates this failure.

3.6. Uncertainty analysis

In this section, we demonstrate and interpret the results of our uncertainty analysis on all three digital pathology datasets.

3.6.1. DigestPath uncertainty maps

Figure 19 provides an illustration of tumor probability and uncertainty maps for a held-out test case from the DigestPath dataset. Figure 19(1,2,3,4,5)(b) corresponds to ground truth image overlayed on the tissue image, while Figure 19(1,2,3,4,5)(c) corresponds to tumor probability heatmap overlayed on tissue

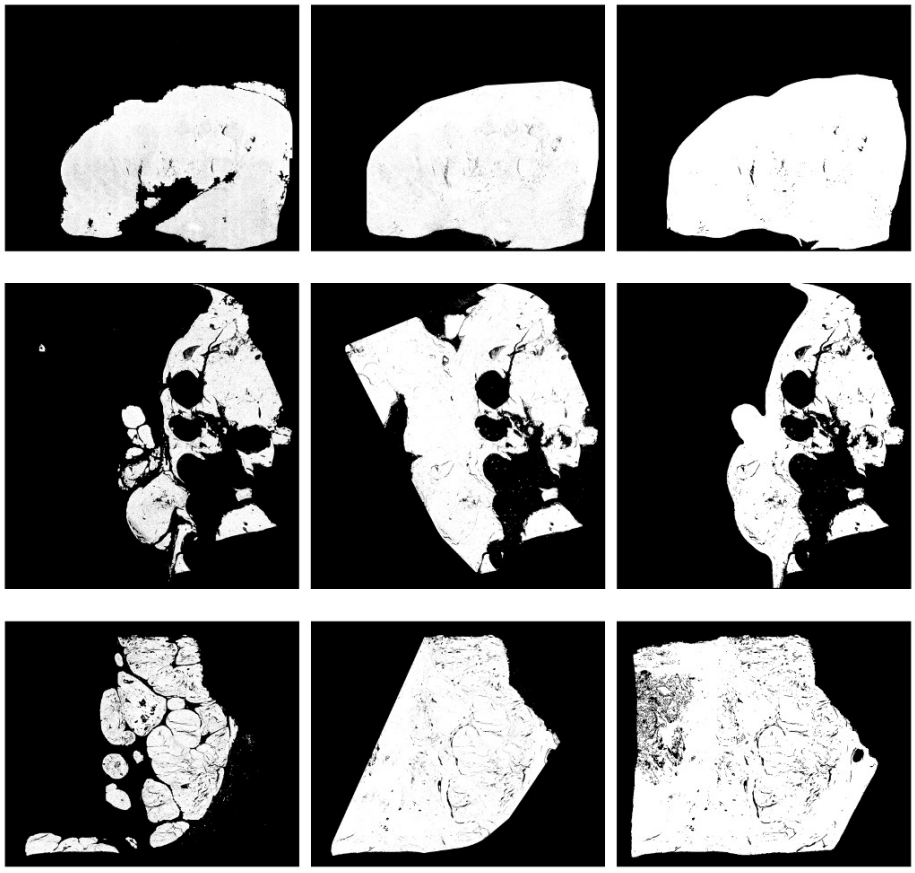


Figure 18: (Left to Right) Viable tumor prediction; Whole tumor prediction; Whole tumor Ground Truth

region. Figures 19(1, 2, 3)(d) describes aleatoric uncertainty with DenseNet, InceptionNet, and DeeplabV3Plus respectively. Since the patches considered for inference in the case of DigestPath were of 256 $\times$ 256, aleatoric uncertainty can be observed in various tumors region. Figure 19(4)(d) describes epistemic uncertainty, where the uncertain region clearly corresponds to the boundary of the tumor tissue region, while Figure 19(5)(d) describes combined uncertainties (average of aleatoric uncertainties for all models along with epistemic uncertainty across the models), which points out various tumorous and boundary regions. In Figure 19(4, 5)(c), we observed ensemble prediction reduced the number of false positives, their by increasing overall dice score to 0.86, which is about 0.04-0.06 of dice score improvement when compared to individual model.

3.6.2. PAIP uncertainty maps

In the Figure 20 provides an illustration of tumor probability and uncertainty maps for a held-out test case from PAIP dataset. Our observations were similar to as discussed in section 3.6.1.

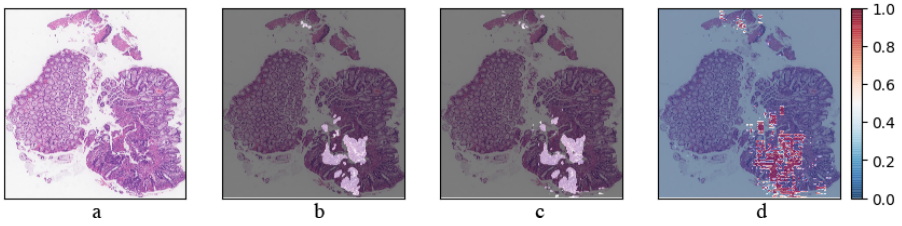
3.6.3. CAMELYON uncertainty maps

In Figure 21 provides an illustration of tumor probability and uncertainty maps for a held-out test case from CAMELYON dataset. Our observations were similar to as discussed in section 3.6.1.

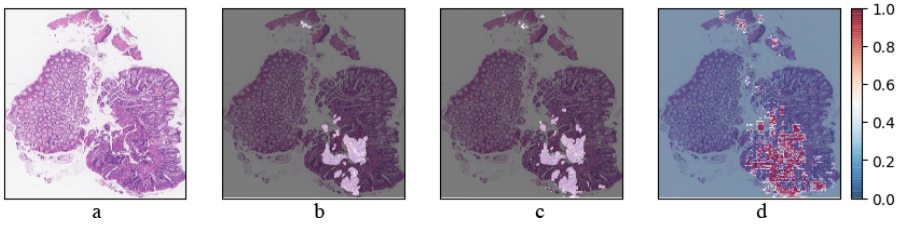
3.7. Transfer learning between cancer sites

To test transfer learning between cancer sites, we first tried inferencing on the DigestPath dataset with a model trained on the CAMELYON dataset. As it can be observed from Figure 22, the model learned inherent tissue concepts that were common between both sites. The similar staining pattern and nuclei similarity could have contributed to this observation.

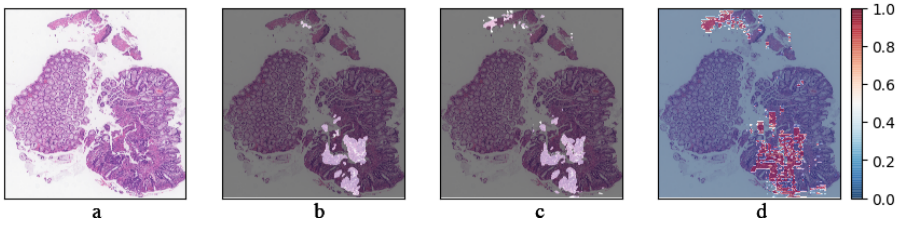
Subsequently, we conducted a comparative analysis on the model weight initialisation. We trained two models on the PAIP dataset; One pre-trained on the CAMELYON dataset and one pre-trained on the Imagenet dataset (Deng et al., 2009). We observed that the former model converged at a faster rate.



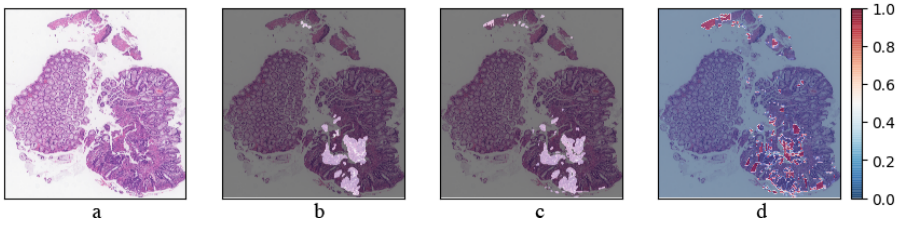
(1) DenseNet model predictions with aleatoric uncertainty (dice = 0.82)



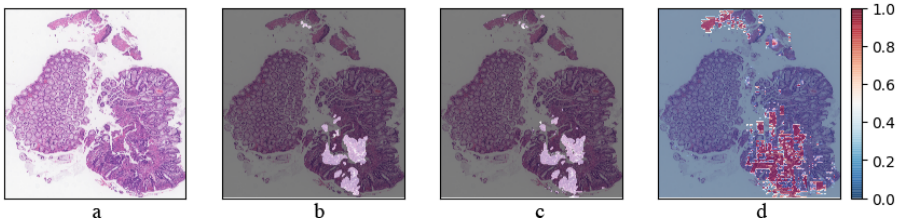
(2) InceptionNet model predictions with aleatoric uncertainty (dice = 0.78)



(3) DeepLabv3Plus model predictions with aleatoric uncertainty (dice = 0.80)

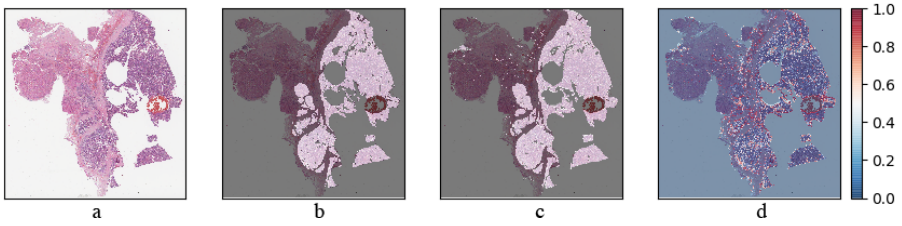


(4) Ensemble Predictions with epistemic uncertainty (dice = 0.86)

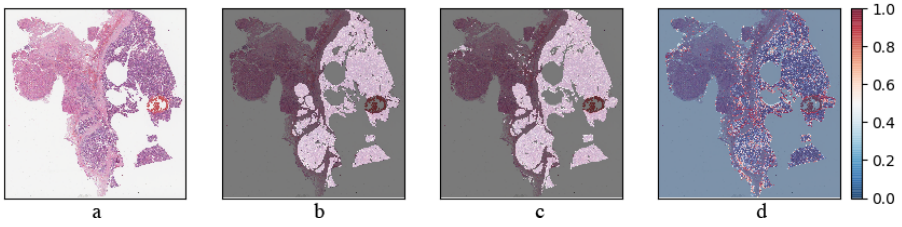


(5) Ensemble predictions with combined uncertainty (dice = 0.86)

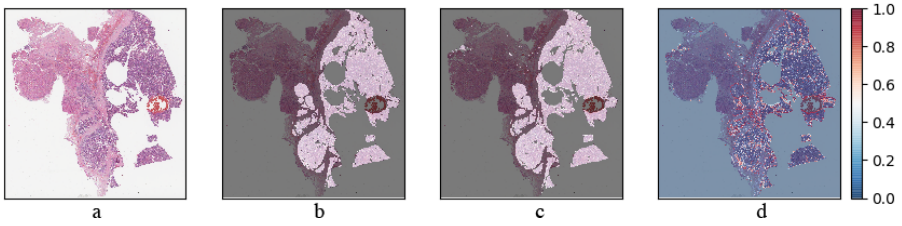
Figure 19: In the figure (a) WSI, (b) Ground truth image overlayed on the tissue region, (c) Tumor probability heatmap overlayed on the tissue region, (d) Corresponding uncertainty map.



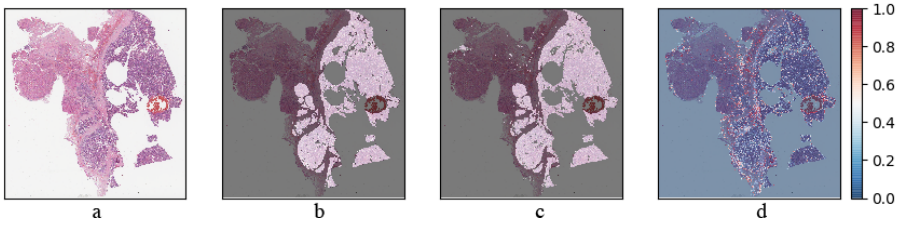
(1) DenseNet model predictions with aleatoric uncertainty (dice = 0.88)



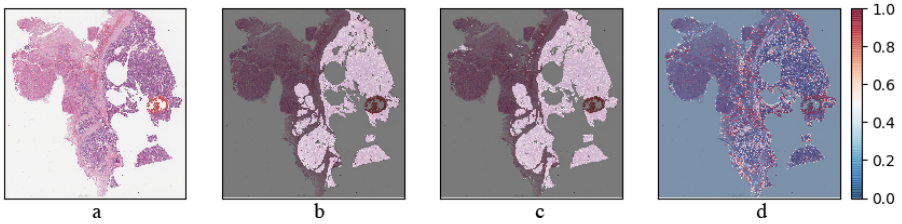
(2) InceptionNet model predictions with aleatoric uncertainty (dice = 0.88)



(3) DeepLabv3Plus model predictions with aleatoric uncertainty (dice = 0.88)

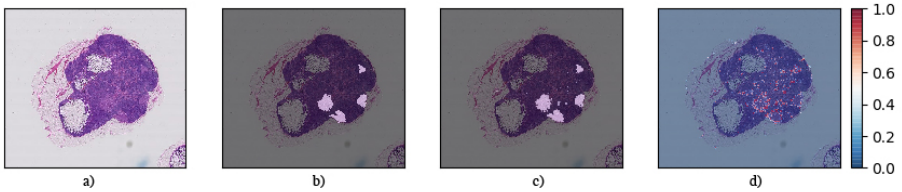


(4) Ensemble Predictions with epistemic uncertainty (dice = 0.91)

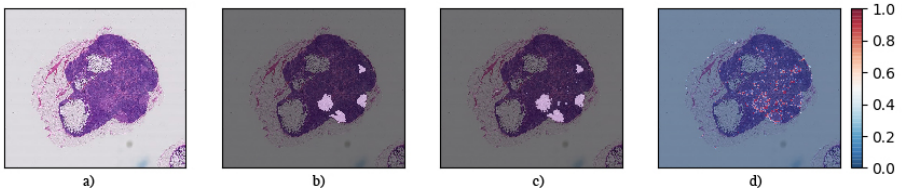


(5) Ensemble predictions with combined uncertainty (dice = 0.91)

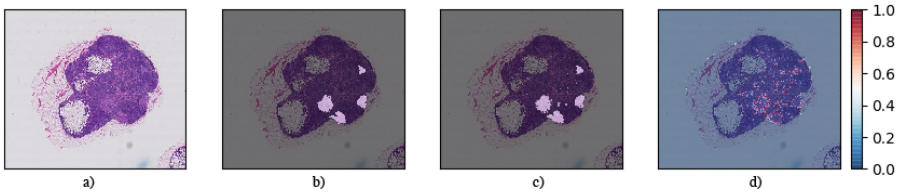
Figure 20: In the figure (a) WSI, (b) Ground truth image overlayed on the tissue region, (c) Tumor probability heatmap overlayed on the tissue region, (d) Corresponding uncertainty map.



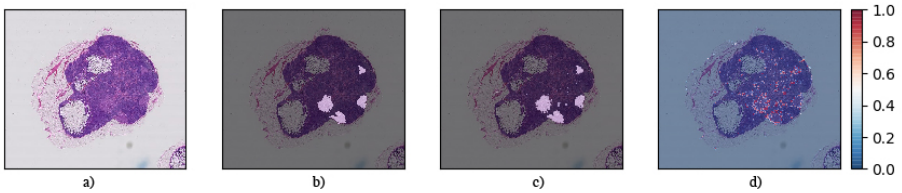
(1) DenseNet model predictions with aleatoric uncertainty (dice = 0.91)



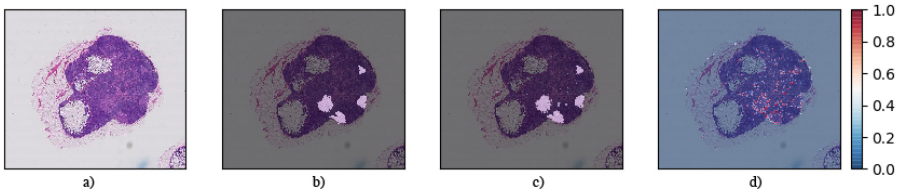
(2) InceptionNet model predictions with aleatoric uncertainty (dice = 0.89)



(3) DeepLabv3Plus model predictions with aleatoric uncertainty (dice = 0.88)



(4) Ensemble Predictions with epistemic uncertainty (dice = 0.94)



(5) Ensemble predictions with combined uncertainty (dice = 0.94)

Figure 21: In the figure (a) WSI, (b) Ground truth image overlayed on the tissue region, (c) Tumor probability heatmap overlayed on the tissue region, (d) Corresponding uncertainty map.

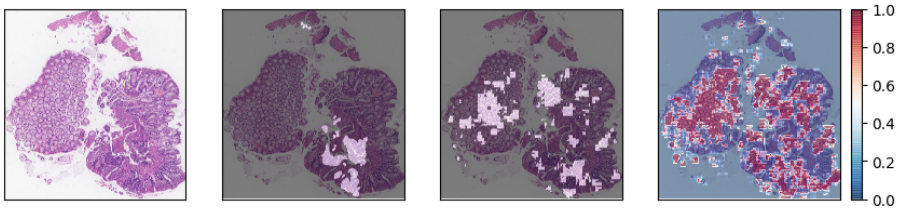


Figure 22: (Left to Right) DigestPath Slide; Ground truth; Tumor prediction with Camelyon model; Aleatoric uncertainty

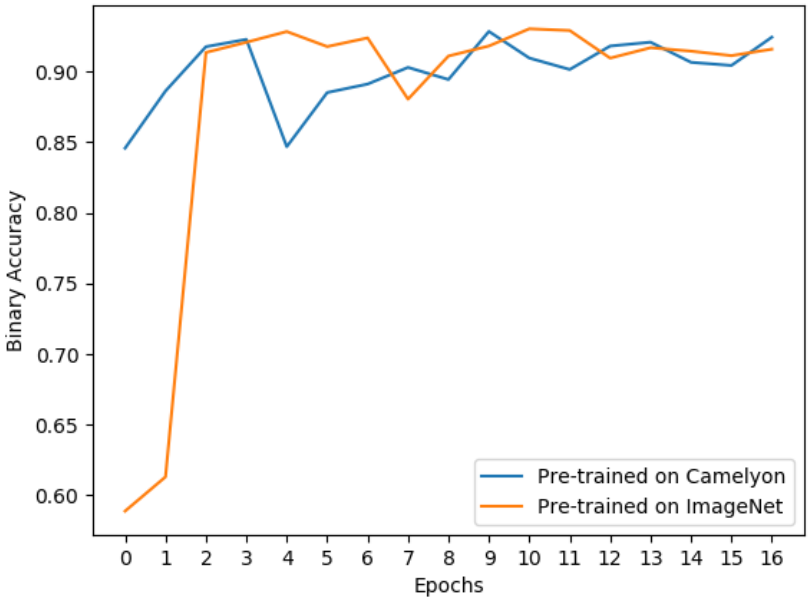


Figure 23: Training loss with Camelyon pre-trained weights and ImageNet pre-trained weights for the PAIP dataset

Figure 23 illustrates this comparison. This observation indicates that models trained on pathology datasets could serve as a better starting point compared to natural image datasets.

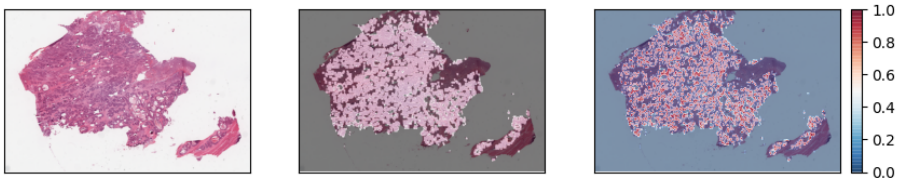
3.8. Model generalizability

To test the generalizability of the model we conducted the model inferences on TCGA colon dataset (gdc), the model was able to accurately classify between tumorous and normal cells as seen in the Figure 24. Even though the model was trained on DigestPath data, our model could generalize well on datasets coming from other sources, Figures 24(1, 2, 3, 4, 5)(b) describes the predicted high probable tumor regions, while in the Figure 24(1, 2, 3, 4, 5)(c) describes corresponding aleatoric and epistemic uncertainties in the prediction. In future, we plan to extend this generalizability study by getting pathologists annotations done and precisely assessing the seg mentation performance.

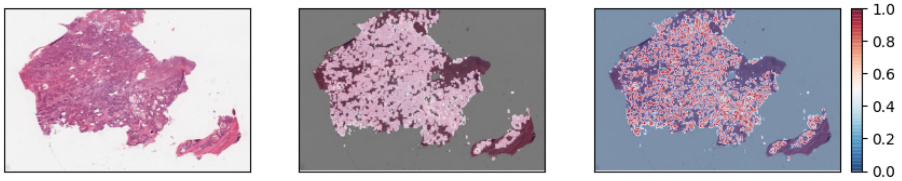
3.9. Hardmining

Our patch extraction scheme from the whole-slide images were based on random uniform sampling. However, this led to some hard examples being excluded from the training set which resulted in the model's poor performance on such regions of WSI. Therefore, we attempted to solve this issue by hardmining the poorly performing regions in WSI and fine-tuning the trained model with this hardmined set.

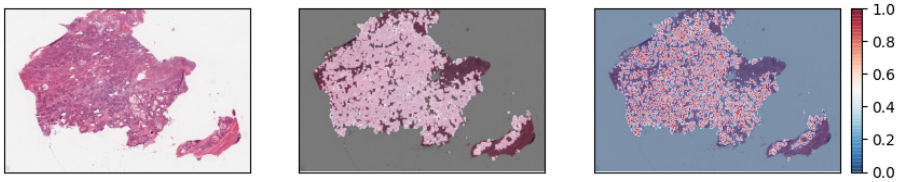
For the experimental analysis with the PAIP dataset, we first extracted 80,000 random patches as the initial training set and trained a model on it. We empirically defined patches that scored less than 0.4 on the Jaccard index as regions were the model performed poorly. We than inferred on the entire dataset and extracted all patches that meet this criterion. In this manner, we obtained a total of 70,000 patches as the hardmined set. We then continued training the model after substituting the training set with the hardmined set. However, at every epoch, we observed the model's accuracy on validation set reduced. We theorized that by training only on the failure cases, the model forgets the initial



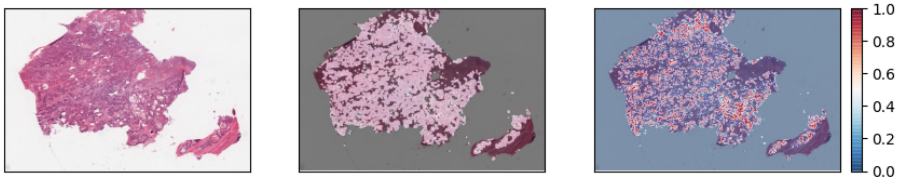
(1) DenseNet model predictions with aleatoric uncertainty



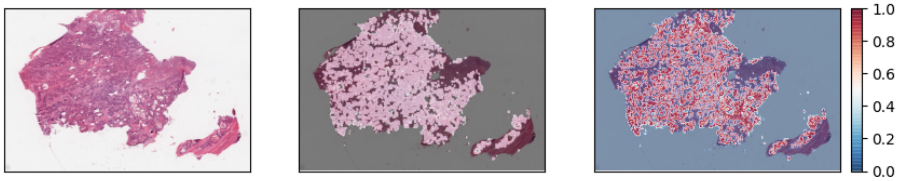
(2) InceptionNet model predictions with aleatoric uncertainty



(3) DeepLabv3Plus model predictions with aleatoric uncertainty



(4) Ensemble Predictions with epistemic uncertainty



(5) Ensemble predictions with combined uncertainty

Figure 24: In the figure (a) WSI, (b) Tumor probability heatmap overlayed on the tissue region, (c) Corresponding uncertainty map.

Table 13: Comparisons with different approaches to automated pN-Staging in CAMELYON17 challenge. Our score reported in the table is from open public leader board. Our best performing method stood rank-3 on the leader-board (Accessed on: 31-Dec-2019). The table shows the performance of individual Random Forest classifiers in the ensemble and RF-Ensemble classifier.

Method	**Cohen Kappa Score**	**Rank**
Lee et al. (2019)	0.9570	1
Pinchaud (2019)	0.9386	2
Ours (RF-Ensemble)	0.9090	3
Ours (RF-PI)	0.8971	12
Ours (RF-PB)	0.9027	9
Ours (RF-CI)	0.8889	18
Ours (RF-CB)	0.9057	6

training set. So, we tried an alternative where we concatenated the patches of both the initial training set and the hard-mined set and trained the model with this combined set. The model's accuracy reduced with this method as well, albeit at a slower rate.

To investigate this occurrence, we examined the patches obtained from hard-mining and analysed them. We found groups of patches that were highly similar to each other. This was because the regions where the model failed were larger in size compared to the dimensions with which the patches were extracted. Therefore, the hardmining algorithm extracts several patches from the same region. As a result, this set was less representative of the data compared to the first set of randomly extracted patches. We observed a similar pattern in the other datasets as well.

Teams	Dice
kuanguang	0.807
zju_realdoctor	0.792
TIA_Lab	0.787
Ours	0.782

Table 14: The table shows the top segmentation scores on DigestPath testset

4. Challenge Results

4.1. Performance on CAMELYON17 Challenge

Table 13 compares the results between our proposed approach and other published results on CAMELYON17 testing dataset (n=500). Our best performing model was RF-Ensemble, which gave a Cohen's kappa score of 0.9090, placing 3rd in Open Leaderboard (out of 120 valid submission entries).

4.2. Performance on DigestPath Challenge

In the case of the DigestPath challenge, we submitted our ensemble model and obtained a Dice score of 0.78 on test set. The scores of all the top-performing teams are tabulated in the Table. 14.

4.3. Performance on PAIP 2019 Challenge

The Table 15 summarises the scores for the top entries. Task 1 is evaluated with the Jaccard metric. For Task 2, first, the absolute accuracy is calculated, and then this score is weighted with the Task 1 score of the corresponding case. For Task 1, all the participants utilized deep learning methods, albeit with different architectures. For Task 2, all the participants used deep learning methods. However, our heuristic based algorithm performed better than most deep learning methods.

Table 15: The table shows the top 5 entries of PAIP 2019. Task 1 corresponds to Viable tumor segmentation and Task 2 corresponds to Viable tumor burden estimation. Note: FNLCR: Frederick National Laboratory for Cancer Research

Team	Task 1	Task 2
FNLCR	0.789	0.752
Sichuan University	0.777	NA
Ours	0.750	0.6337
Alibaba	0.672	0.6199
Sejong University	0.665	0.6330

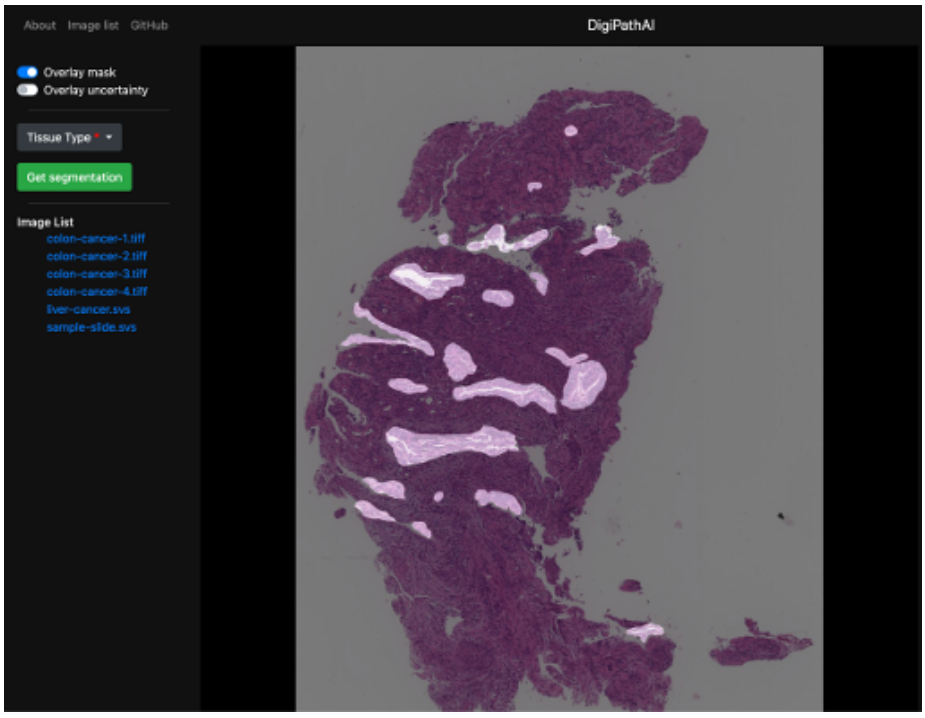


Figure 25: User Interface of the software

5. Open Source Contribution

We developed an open-source application (Haran Rajkumar, 2019) on top of our segmentation pipeline. The application 25 can load WSI images, perform segmentation, and calculate the uncertainties. It features an API with which researchers can utilize the segmentation pipeline within their applications. Conversely, the application's modular structure allows for researchers to test their segmentation pipeline with the application's GUI as well. The slide viewer was built using OpenSlide (Goode et al., 2013) and OpenSeadragon (Antoine Vandecreme).

6. Discussion and Conclusion

We developed an automated end-to-end framework for tumor tissue segmentation and whole slide image analysis that showed state-of-the-art results on three publicly available histopathology challenges. We approached the problem of segmentation of gigapixel WSI images using the divide and conquer strategy by dividing the large image into computationally feasible patch sizes and running segmentation algorithms on patches and stitching segmented patches to generate the segmentation of the entire WSI. We approached the problem of patch image segmentation using fully convolutional neural networks (FCN). FCNs are encoder-decoder based architectures employed for generating dense pixel-level classification. We used some of the state-of-the-art CNNs on natural image tasks as encoders for our FCNs, and the decoder was basically a learnable upsampling module to generate dense predictions. Our segmentation framework is an ensemble comprising of multiple FCN architectures, each independently trained on different subsets of the training data. The ensemble generated the tumor probability map by averaging the posterior probability maps of all the FCNs. The ensemble approach showed superior segmentation performance when compared to its individual constituting FCNs. The patch-based segmentation methods for large-sized images suffer from loss of neighboring context information at patch borders. We attempted to address this issue during inference by using a larger

patch size and averaging overlapping patch regions posterior probability maps while stitching tumor probability maps for WSI. Depending upon tissue area covered on glass-slide, our inference pipeline takes about approximately 30-75 minutes for generating the entire tumor probability map. In addition to the generation of tumor heat-map, we also incorporated a methodology for generating uncertainty maps based on model and data variability. These uncertainty maps would assist in better interpretation by pathologists and fine-tuning the model with uncertain regions. Our proposed hard-mining approach showed decreased segmentation performance because of the adopted fine-tuning procedure on the trained model. Hence, one of the directions of our future work would involve developing effective methodologies for the online extraction of good representative patches from WSI during the training phase rather than the extraction of a fixed set of random patches(Lee et al., 2019; Pinchaud, 2019). Other areas to explore would be in the design of efficient and multi-resolution FCN architectures for capturing multi-resolution information from WSI images (Graham et al., 2019). Our experimental results on transfer learning showed that pre-training models with different histopathology datasets could act as good starting points for training models were pathology datasets are limited. Post-processing techniques could be one of the directions to improve generated WSI segmentation, techniques such as patch-based conditional random fields (Krähenbühl and Koltun, 2011; Li and Ping, 2018) could be employed to refine the generated segmentation masks rather than employing hardcoded threshold values. The segmentation of WSIs is usually the primary step for many analysis tasks such as metastases classification and estimation of tumor burden. In this regard, we developed an automated pipeline for lymph node metastases classification and pN-staging. We proposed an ensemble of multiple Random Forest classifiers, each trained on different subsets of the training data. The training data was prepared by extracting meaningful features from a pathologist's viewpoint from the tumor probability maps. We also demonstrated the efficacy of our synthetic samples in addressing class imbalance datasets for such classification tasks.

Our method for tumor burden estimation used an empirical method for

estimating the whole tumor region. However, it still performed on par with other deep learning approaches that modeled it as a segmentation problem. This shows that the convex hull is a good approximation and also computationally inexpensive. This method could be developed further by incorporating learning-based methods. For example, the convex hull output could be used as an initial point for active contours-based models (Kass et al., 1988).

7. Author Contributions

MK, HR and AK developed the generalised pipeline together. MK, HR and AK experimented on CAMELYON, PAIP and DigestPath respectively. MK supervised HR and AK on the experimentation. HR and AK developed the open-source software together. MK, HR and AK wrote and revised the manuscript. GK and BS edited the manuscript, supervised and funded the study.

Appendix A. Evaluation metrics

Appendix A.0.1. Dice Coefficient

The Dice is a metric used to measure the overlap between two given sets. In the case of segmentation, dice coefficient measures the overlap between the proposed model prediction(P) and the expert pathologists' ground truth (GT). This was the evaluation metric used in the DigestPath19 challenge. Mathematically, dice coefficient can be expressed as equation A.1.

$$Dice(P, GT) = \frac{2 * |P \cap GT|}{|P \cup GT|} \tag{A.1}$$

Appendix A.0.2. Jaccard Coefficient

The Jaccard metric, similar to the Dice coefficient, measures the intersection over union between two given sets. For the task of segmentation, Jaccard coefficient measures overlap between prediction (P) and expert pathologists' ground truth (GT). This was the evaluation metric used in the PAIP 2019 (PAIP, 2019) challenge. Mathematically, Jaccard coefficient can be expressed as equation A.2.

$$Jaccard(P, GT) = \frac{|P \cap GT|}{|P \cup GT|} \tag{A.2}$$

Appendix A.0.3. FROC

One of the metrics used in CM16 challenge for lesion-based evaluation was free-response receiver operating characteristic (FROC) curve. The FROC curve was defined as the plot of sensitivity versus the average number of false-positives per image. We used CM16 challenge testing dataset for evaluating the performance of our algorithms for lesion detection/localization. The detection/localization performance was summarized using Free Response Operating Characteristic (FROC) curves. This was similar to ROC analysis, except that the false positive rate on the x-axis is replaced by the average number of false positives per WSI.

In the CM16 challenge, a true positive was considered, if the location of the detected region was within the annotated ground truth lesion.

- If there were multiple findings for a single ground truth region, they were counted as a single true positive finding and none of them were counted as a false positive.
- All detections that were not within a specific distance from the ground truth annotations were counted as false positives.

The final FROC score was defined as the average sensitivity at 6 predefined false positive rates: 1/4, 1/2, 1, 2, 4, and 8 FPs per whole slide image.

Appendix A.0.4. Cohen's kappa score

Cohen's kappa (Fleiss and Cohen, 1973) is a statistic that measures the inter-rater reliability for categorical variables. In CM17 challenge for evaluating pN-staging of the patients the metric used was Cohen's kappa with five classes and quadratic weights. The kappa metric ranges from -1 to +1, where 1 represented perfect agreement with the raters, and 0 represented the amount of agreement that can be expected by random chance and, a negative value represented lower than chance agreement.

References

URL: https://portal.gdc.cancer.gov/repository?facetTab=cases&filters={"op":"and","content":[{"op":"in","content":{"field":"cases.project.program.name","value":["TCGA"]}},{"op":"in","content":{"field":"cases.project.project_id","value":["TCGA-COAD"]}}]}.

The camelyon16 challenge. 2017a. URL: https://camelyon16.grand-challenge.org/; accessed: 31- Dec- 2019.

The camelyon17 challenge. 2017b. URL: https://camelyon17.grand-challenge.org/; accessed: 31- Dec- 2019.

Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:160304467 2016;.

Amin MB, Edge SB. AJCC cancer staging manual. Springer, 2017.

Antoine Vandecreme Ian Gilman MSCT. Openseadragon. URL: http://openseadragon.github.io; accessed: 31- Dec- 2019.

Arvaniti E, Fricker KS, Moret M, Rupp N, Hermanns T, Fankhauser C, Wey N, Wild PJ, Rueschoff JH, Claassen M. Automated gleason grading of prostate cancer tissue microarrays via deep learning. Scientific reports 2018;8.

Bagari A, Kumar A, Kori A, Khened M, Krishnamurthi G. A combined radio-histological approach for classification of low grade gliomas. In: International MICCAI Brainlesion Workshop. Springer; 2018. p. 416–27.

Bakas S, Reyes M, Jakab A, Bauer S, Rempfler M, Crimi A, Shinohara RT, Berger C, Ha SM, Rozycki M, et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall

survival prediction in the brats challenge. arXiv preprint arXiv:181102629 2018;.

Bandi P, Geessink O, Manson Q, Van Dijk M, Balkenhol M, Hermsen M, Bejnordi BE, Lee B, Paeng K, Zhong A, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. IEEE transactions on medical imaging 2018;38(2):550–60.

Batista GE, Prati RC, Monard MC. A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD explorations newsletter 2004;6(1):20–9.

Begoli E, Bhattacharya T, Kusnezov D. The need for uncertainty quantification in machine-assisted medical decision making. Nature Machine Intelligence 2019;1(1):20.

Bejnordi BE, Veta M, Van Diest PJ, Van Ginneken B, Karssemeijer N, Litjens G, Van Der Laak JA, Hermsen M, Manson QF, Balkenhol M, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. Jama 2017;318(22):2199–210.

Bychkov D, Linder N, Turkki R, Nordling S, Kovanen PE, Verrill C, Walliander M, Lundin M, Haglund C, Lundin J. Deep learning based tissue analysis predicts outcome in colorectal cancer. Scientific reports 2018;8(1):3395.

Chaudhary K, Poirion OB, Lu L, Garmire LX. Deep learning–based multi-omics integration robustly predicts survival in liver cancer. Clinical Cancer Research 2018;24(6):1248–59.

Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research 2002;16:321–57.

Chen LC, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:170605587 2017;.

Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV). 2018. p. 801–18.

Chuang SC, La Vecchia C, Boffetta P. Liver cancer: descriptive epidemiology and risk factors other than hbv and hcv infection. Cancer letters 2009;286(1):9–14.

Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. Ieee; 2009. p. 248–55.

Diamond J, Anderson NH, Bartels PH, Montironi R, Hamilton PW. The use of morphological characteristics and texture analysis in the identification of tissue composition in prostatic neoplasia. Human pathology 2004;35(9):1121–31.

Elmore JG, Longton GM, Carney PA, Geller BM, Onega T, Tosteson AN, Nelson HD, Pepe MS, Allison KH, Schnitt SJ, et al. Diagnostic concordance among pathologists interpreting breast biopsy specimens. Jama 2015;313(11):1122–32.

Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (voc) challenge. International journal of computer vision 2010;88(2):303–38.

Fleiss JL, Cohen J. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. Educational and psychological measurement 1973;33(3):613–9.

Fleming M, Ravula S, Tatishchev SF, Wang HL. Colorectal carcinoma: Pathologic aspects. Journal of gastrointestinal oncology 2012;3(3):153.

Gal Y, Ghahramani Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: international conference on machine learning. 2016. p. 1050–9.

Gibson E, Robu MR, Thompson S, Edwards PE, Schneider C, Gurusamy K, Davidson B, Hawkes DJ, Barratt DC, Clarkson MJ. Deep residual networks for automatic segmentation of laparoscopic videos of the liver. In: Medical Imaging 2017: Image-Guided Procedures, Robotic Interventions, and Modeling. International Society for Optics and Photonics; volume 10135; 2017. p. 101351M.

Giuliano AE, Ballman KV, McCall L, Beitsch PD, Brennan MB, Kelemen PR, Ollila DW, Hansen NM, Whitworth PW, Blumencranz PW, et al. Effect of axillary dissection vs no axillary dissection on 10-year overall survival among women with invasive breast cancer and sentinel node metastasis: the acosog z0011 (alliance) randomized clinical trial. Jama 2017;318(10):918–26.

Giuliano AE, Hunt KK, Ballman KV, Beitsch PD, Whitworth PW, Blumencranz PW, Leitch AM, Saha S, McCall LM, Morrow M. Axillary dissection vs no axillary dissection in women with invasive breast cancer and sentinel node metastasis: a randomized clinical trial. Jama 2011;305(6):569–75.

Goode A, Gilbert B, Harkes J, Jukic D, Satyanarayanan M. Openslide: A vendor-neutral software foundation for digital pathology. Journal of pathology informatics 2013;4.

Graham S, Chen H, Gamper J, Dou Q, Heng PA, Snead D, Tsang YW, Rajpoot N. Mild-net: Minimal information loss dilated network for gland instance segmentation in colon histology images. Medical image analysis 2019;52:199–211.

Greff K, Srivastava RK, Koutník J, Steunebrink BR, Schmidhuber J. Lstm: A search space odyssey. IEEE transactions on neural networks and learning systems 2016;28(10):2222–32.

Guray M, Sahin AA. Benign breast diseases: classification, diagnosis, and management. The oncologist 2006;11(5):435–49.

Gurcan MN, Boucheron L, Can A, Madabhushi A, Rajpoot N, Yener B. Histopathological image analysis: A review. IEEE reviews in biomedical engineering 2009;2:147.

Hamilton S. Carcinoma of the colon and rectum. World health organization classification of tumors Pathology and genetics of tumors of the digestive system 2000;:105–19.

Haran Rajkumar AK. Digipathai. 2019. URL: `https://github.com/haranrk/DigiPathAI`; accessed: 31- Dec- 2019.

Hawkes N. Cancer survival data emphasise importance of early diagnosis. 2019.

He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. p. 770–8.

Iandola F, Moskewicz M, Karayev S, Girshick R, Darrell T, Keutzer K. Densenet: Implementing efficient convnet descriptor pyramids. arXiv preprint arXiv:14041869 2014;.

ICAIR . Icair 2018 - breast cancer histology images. 2019. URL: `https://iciar2018-challenge.grand-challenge.org/`; accessed: 31- Dec- 2019.

Kaluva KC, Khened M, Kori A, Krishnamurthi G. 2d-densely connected convolution neural networks for automatic liver and tumor segmentation. arXiv preprint arXiv:180202182 2018;.

Kass M, Witkin A, Terzopoulos D. Snakes: Active contour models. International journal of computer vision 1988;1(4):321–31.

Kather JN, Krisam J, Charoentong P, Luedde T, Herpel E, Weis CA, Gaiser T, Marx A, Valous NA, Ferber D, et al. Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. PLoS medicine 2019;16(1):e1002730.

Kendall A, Gal Y. What uncertainties do we need in bayesian deep learning for computer vision? In: Advances in neural information processing systems. 2017. p. 5574–84.

Kermany DS, Goldbaum M, Cai W, Valentim CC, Liang H, Baxter SL, McKeown A, Yang G, Wu X, Yan F, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell 2018;172(5):1122–31.

Krähenbühl P, Koltun V. Efficient inference in fully connected crfs with gaussian edge potentials. In: Advances in neural information processing systems. 2011. p. 109–17.

Lee B, Paeng K. A robust and effective approach towards accurate metastasis detection and pn-stage classification in breast cancer. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer; 2018. p. 841–50.

Lee S, Oh S, Choi K, Kim SW. Automatic classification on patient-level breast cancer metastases 2019;URL: `https://camelyon17.grand-challenge.org/media/evaluation-supplementary/80/22149/46fc579c-51f0-40c4-bd1a-7c28e8033f33/Camelyon17_.pdf`; accessed: 31-Dec-2019.

Leibig C, Allken V, Ayhan MS, Berens P, Wahl S. Leveraging uncertainty information from deep neural networks for disease detection. Scientific reports 2017;7(1):17816.

Li J, Yang S, Huang X, Da Q, Yang X, Hu Z, Duan Q, Wang C, Li H. Signet ring cell detection with a semi-supervised learning framework. In: International Conference on Information Processing in Medical Imaging. Springer; 2019. p. 842–54.

Li W, Jia F, Hu Q. Automatic segmentation of liver tumor in ct images with deep convolutional neural networks. Journal of Computer and Communications 2015;3(11):146.

Li Y, Ping W. Cancer metastasis detection with neural conditional random field. In: Medical Imaging with Deep Learning. 2018. .

Liaw A, Wiener M, et al. Classification and regression by randomforest. R news 2002;2(3):18–22.

Litjens G, Bandi P, Ehteshami Bejnordi B, Geessink O, Balkenhol M, Bult P, Halilovic A, Hermsen M, van de Loo R, Vogels R, et al. 1399 h&e-stained sentinel lymph node sections of breast cancer patients: the camelyon dataset. GigaScience 2018;7(6):giy065.

Litjens G, Sánchez CI, Timofeeva N, Hermsen M, Nagtegaal I, Kovacs I, Hulsbergen-Van De Kaa C, Bult P, Van Ginneken B, Van Der Laak J. Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. Scientific reports 2016;6:26286.

Liu Y, Gadepalli K, Norouzi M, Dahl GE, Kohlberger T, Boyko A, Venugopalan S, Timofeev A, Nelson PQ, Corrado GS, et al. Detecting cancer metastases on gigapixel pathology images. arXiv preprint arXiv:170302442 2017;.

Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015. p. 3431–40.

Madabhushi A, Lee G. Image analysis and machine learning in digital pathology: Challenges and opportunities. 2016.

Malhotra GK, Zhao X, Band H, Band V. Histological, molecular and functional subtypes of breast cancers. Cancer biology & therapy 2010;10(10):955–60.

Nanthagopal AP, Rajamony RS. Classification of benign and malignant brain tumor ct images using wavelet texture parameters and neural network classifier. Journal of visualization 2013;16(1):19–28.

Otsu N. A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics 1979;9(1):62–6.

Paeng K, Hwang S, Park S, Kim M. A unified framework for tumor proliferation score prediction in breast histopathology. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Springer; 2017. p. 231–9.

PAIP . Paip 2019 - liver cancer segmentation. 2019. URL: https://paip2019.grand-challenge.org; accessed: 31- Dec- 2019.

Petushi S, Garcia FU, Haber MM, Katsinis C, Tozeren A. Large-scale computations on histology images reveal grade-differentiating parameters for breast cancer. BMC medical imaging 2006;6(1):14.

Pinchaud N. Camelyon17 grand challenge. 2019. URL: https://camelyon17.grand-challenge.org/media/evaluation-supplementary/80/26459/345cb218-5d96-4125-80ce-e1b12cd64c7a/Camelyon17_submission.pdf; accessed: 31-Dec-2019.

Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, Ding D, Bagul A, Langlotz C, Shpanskaya K, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv preprint arXiv:171105225 2017;.

Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. Springer; 2015. p. 234–41.

Salamat MS. Robbins and cotran: Pathologic basis of disease. 2010.

Shapcott CM, Rajpoot N, Hewitt K. Deep learning with sampling for colon cancer histology images. Frontiers in Bioengineering and Biotechnology 2019;7:52.

Sirinukunwattana K, e Ahmed Raza S, Tsang YW, Snead DR, Cree IA, Rajpoot NM. Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. IEEE Trans Med Imaging 2016;35(5):1196–206.

Sites A. Seer cancer statistics review 1975-2011. Bethesda, MD: National Cancer Institute 2014;.

Sobin LH, Gospodarowicz MK, Wittekind C. TNM classification of malignant tumours. John Wiley & Sons, 2011.

Stewart B, Wild CP, et al. World cancer report 2014 2019;.

Szegedy C, Ioffe S, Vanhoucke V, Alemi AA. Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-First AAAI Conference on Artificial Intelligence. 2017. .

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2015. p. 1–9.

Tizhoosh HR, Pantanowitz L. Artificial intelligence and digital pathology: Challenges and opportunities. Journal of pathology informatics 2018;9.

Tomek I. Two modifications of cnn. IEEE Trans Systems, Man and Cybernetics 1976;6:769–72.

Wählby C, Sintorn IM, Erlandsson F, Borgefors G, Bengtsson E. Combining intensity, edge and shape information for 2d and 3d segmentation of cell nuclei in tissue sections. Journal of microscopy 2004;215(1):67–76.

van der Walt S, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, Gouillart E, Yu T, the scikit-image contributors . scikit-image: image processing in Python. PeerJ 2014;2:e453. URL: `https://doi.org/10.7717/peerj.453`. doi:`10.7717/peerj.453`.

Wang D, Khosla A, Gargeya R, Irshad H, Beck A. Deep learning for identifying metastatic breast cancer (2016). arXiv preprint arXiv:160605718 2018;.

Weng SF, Reps J, Kai J, Garibaldi JM, Qureshi N. Can machine-learning improve cardiovascular risk prediction using routine clinical data? PloS one 2017;12(4):e0174944.

Wolberg WH, Street WN, Mangasarian O. Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates. Cancer letters 1994;77(2-3):163–71.

Wu K, Chen X, Ding M. Deep learning based classification of focal liver lesions with contrast-enhanced ultrasound. Optik-International Journal for Light and Electron Optics 2014;125(15):4057–63.

Xu J, Luo X, Wang G, Gilmore H, Madabhushi A. A deep convolutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images. Neurocomputing 2016;191:214–23.

Yasaka K, Akai H, Abe O, Kiryu S. Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced ct: a preliminary study. Radiology 2017;286(3):887–96.