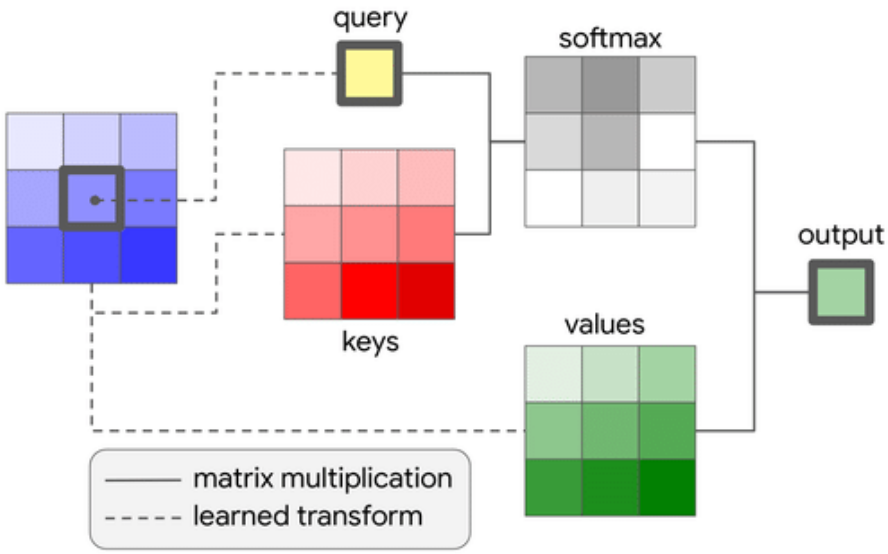
COLLEGE OF ENGINEERING

Preliminary Exam Report:

Applications of Deep Learning to Cancer Detection in Digital Pathology

Somayeh Seifi Shalamzari
Department of Electrical and Computer Engineering
College of Engineering, Temple University
1947 North 12th Street, Philadelphia, Pennsylvania 19122
Email: Somayeh.seifi.shalamzari@temple.edu

April 25, 2023



Examining Committee:

Dr. Joseph Picone, Committee Chair, Department of Electrical and Computer Engineering, College of Engineering, Temple University

Dr. Fouzia Ahmad, Committee Member, Department of Electrical and Computer Engineering, College of Engineering, Temple University

Dr. Iyad Obeid, Committee Member, Department of Electrical and Computer Engineering, College of Engineering, Temple University

Dr. Nancy Pleshko, Committee Member, Department of Bioengineering, College of Engineering, Temple University

Dr. Eduard Dragut, Committee Member, Department of Computer and Information Sciences, College of Science and Technology, Temple University

EXECUTIVE SUMMARY

Digital pathology has become a field of great interest in recent years because deep learning is enabling the development of a new generation of technology whose performance rivals that of pathologists using a manual review process involving analog microscopes. Deep learning systems can help reduce a pathologist's workload by providing decision support. The declining number of physicians pursuing careers in pathology necessitates the development of tools to increase efficiency and productivity. In this report, we discuss the essential components of deep learning systems that analyze histopathological images. We specifically focus on cancer detection from whole slide images (WSI).

Working with whole slide images, which are extremely high resolution (e.g., $100k \times 100k$ pixels), is a significant computational challenge. A large database of these images, which is required to perform state-of-the-art deep learning, cannot fit into a computer's physical memory. Further, to provide useful diagnostic information for pathologists, these images must be segmented into small patches and analyzed patch by patch. This allows the system to identify local regions in each slide that supported the diagnosis but also risks a significant increase in the false alarm rate, which has important clinical implications. There are three fundamental challenges one faces in applying deep learning to digital pathology: (1) creation of a scientifically sound evaluation paradigm, (2) design of suitable pre- and postprocessing of the data to make it suitable for machine learning research, and (3) implementation of an architecture that can successfully classify small regions, or patches, of high-resolution images. We have selected three papers that address each one of these challenges.

The first paper, titled "Artificial Intelligence for Diagnosis and Gleason Grading of Prostate Cancer: The PANDA Challenge," presents a community-wide evaluation on data that included over $10{,}000$ training images and over $1{,}000$ evaluation images from five institutions. A total of $34{,}262$ algorithms submitted by $1{,}290$ developers were initially evaluated. Fifteen leading systems were shown to achieve a similar level of agreement with decisions made by a committee of pathologists and uropathologists (quadratically weighted kappa score of 0.828 at a 95% confidence interval). These algorithms performed similarly on blind evaluation data, supporting the hypothesis that differences between algorithms are small compared to learning how to make effective use of data.

The second paper, titled "A generalized deep learning framework for whole-slide image segmentation and analysis," discusses the segmentation process. The authors used an ensemble segmentation model that divides the image into a series of smaller patches and averages decisions that came from a pool of well-known segmentation algorithms. Four popular pathology databases were used to evaluate the approach. The proposed algorithm achieved a kappa score of 0.91 on the CAMELYON17 test set ($N = 500$) and a Dice similar coefficient (DSC) of 0.78 on the DigestPath test set ($N = 212$). The combination of a kappa score above 0.8 and a DSC above 0.7 indicates strong agreement.

The third paper, titled "Contextual Transformer Networks for Visual Recognition," (COTNet) introduces a robust model with low complexity that uses a transformer-like approach that integrates convolutional and attention mechanisms. The authors evaluated their approach on object detection, semantic segmentation, and instance segmentation, and showed that their approach reduced complexity and achieved better performance than contemporary approaches based on deep convolutional networks with residual interconnections. For example, COTNet achieved a top-1 accuracy of 80%, which was 1.5% absolute higher than ResNet-101, while reducing complexity by 1.8 GFLOPS and $6.3M$ parameters.

Since a cancer diagnosis is a life-changing event, any clinical system must have a low false alarm rate. This necessitates a deep learning system that can robustly segment data. In this report, we have introduced an experimental paradigm and some associated algorithms that offer the potential for automating the diagnosis process, thereby allowing pathologists to spend time on the most significant images. These algorithms can provide valuable decision support for pathologists.

1. INTRODUCTION

Histopathology is the study of stained tissue slides for cancer diagnosis (Gurcan et al., 2009). The analysis of histopathology images is a complex and time-consuming process that requires highly trained and knowledgeable pathologists. The increasing incidence of cancer cases coupled with a shortage of trained specialists has made early diagnosis of cancer a challenging task. Furthermore, the existence of varying levels of discordance among pathologists in cancer diagnosis underscores the necessity of employing computer-aided techniques to assist pathologists (Elmore et al., 2015). Recent advancements in whole-slide imaging (WSI) technology and deep learning methods due to the availability of computational power have opened up new avenues for the early detection of cancer (Litjens et al., 2017). However, the high resolution of WSIs and the lack of annotated images present significant challenges for the efficient processing of these images using deep learning techniques. To overcome this challenge, image segmentation techniques can be applied to detect cancerous cells in histopathology images. Image segmentation involves dividing an image into multiple regions or segments. The lack of reproducibility, which, stems from the inadequate independent validation data is a significant challenge faced by deep learning algorithms. In addition to addressing the aforementioned limitations, it is imperative to introduce a deep network that is both computationally and practically effective. Achieving this objective can facilitate the reproducibility of deep learning algorithms and enhance their overall efficacy in various applications.

Prior to the emergence of deep learning, machine learning algorithms relied heavily on feature engineering. The accuracy of classification was therefore directly related to the quality of the extracted features. The advent of deep learning has been advantageous due to its ability to automate feature extraction (Cruz-Roa et al., 2014). Deep learning models possess a remarkable capacity to extract complex features and subsequently perform downstream tasks such as classification, rendering them particularly attractive to machine learning scientists in the field of WSI. Figure 1 shows an annotated whole slide image. A framework for segmentation based on ensembles has been proposed by Hameed et al. (2020). Qin et al. (2018) drove patch samples and fed them to the multilevel feature pyramid, then performed a multiclass features pyramid to drive semantic segmentation. For fine segmentation, Guo et al. (2019) used a combination of Inception-v3 and a cascaded deep convolutional network. To segment breast cancer images, Priego-Torres et al. (2020) proposed a method based on patch extraction. In his method, patches are extracted from the entire image and then combined using fully connected conditional random fields. This approach offers a promising segmentation pipeline for the analysis of breast cancer images. Roy et al. (2021) proposed a deep learning approach that utilizes multiple resolutions and a customized reconstruction loss to achieve viable tumor segmentation in liver WSIs. This method provides a promising solution for the accurate segmentation of liver tumors in WSI.

The potential of deep learning algorithms in cancer detection is widely recognized; however, their efficacy in practice is hindered by the lack of reproducibility. The requirement for a diverse and unbiased validation set is a key factor contributing to this issue. The inherent susceptibility of deep learning algorithms (Nagendran et al., 2020). The absence of reproducibility poses significant challenges to the widespread implementation of these algorithms.

Figure 1. An annotated breast tissue

Accurate and reliable computer-aided systems for cancer detection are critical due to the severe consequences of diagnostic errors. To mitigate this, it is recommended to use blind validation sets from diverse locations and patient populations. Machine learning algorithms should also be designed to be generalizable across datasets, rather than tailored to specific ones (Maier-Hein et al., 2018). Such AI systems can assist pathologists in accurately and promptly detecting cancer tissue, as well as identifying cases of high severity. To develop such systems, international competitions can be organized that involve participants, datasets, and experts from diverse geographic regions. This would enable the development of robust and accurate computer-aided systems for cancer detection that are effective across different populations and clinical settings.

The accurate classification of small patches in WSIs requires a robust neural network. Convolutional neural networks (CNNs) have been widely employed for visual downstream tasks for several years, owing to their ability to extract complex features automatically using mathematical techniques such as convolution (Simonyan & Zisserman, 2014; Szegedy et al., 2015; Tan & Le, 2019). Yan Lecun and colleagues first introduced convolutional methods with their LenNet-5 network, which was designed for handwritten character recognition and consisted of two convolutional and pooling layers, followed by a fully connected layer and softmax (Lecun et al., 1998). In 2012, Alex Krizhevsky and his team presented AlexNet, which won the ImageNet competition. While the architecture of AlexNet resembled that of LeNet, it had deeper layers. However, its heavy hyper parameterization posed significant limitations (Krizhevsky et al., 2012). VGG Net subsequently addressed these limitations by substituting smaller kernel-sized filters for larger ones (such as 11 and 5 in the first and second convolutional layers) (Simonyan & Zisserman, 2015). Nevertheless, VGG Net suffered from a vanishing gradient problem that was resolved by He et al. (2016) through the introduction of ResNet, which went on to win the ILSVRC-2015 competition. Residual connections were applied to the ResNet, which effectively solved the vanishing gradient problem (He et al., 2016).

The CCN-based networks have shown promise, but they also have some disadvantages. CNNs use locally applied filters in convolution layers. Since these filters extract only local features, CNNs have limitations when it comes to processing long-range dependencies between pixels. Self-attention-based transformer architecture has emerged as a breakthrough in the field of natural language processing (Vaswani et al., 2017). This approach allows for the consideration of long-term dependencies among input tokens, which has attracted significant attention from the computer vision community. Dosovitskiy et al. (2021) proposed the Vision Transformer (Vit) as a novel approach that utilizes transformer encoders for computer vision. To structure the input image data in a manner similar to natural language processing, the authors divided the images into $16x16$ patches before feeding them into the encoder. Attention weights can help the model understand complicated input features. Furthermore, self-attention can capture the entire image at once, unlike convolutional neural networks (CNNs) (Dai et al., 2021). However, calculating the dependencies between each pixel and the entire image can be computationally expensive for images of the size of whole-slide images (WSI). This has led to growing interest in recent years in hybrid methods that combine convolution and attention mechanisms (Dai et al., 2021). Hybrid models rely on convolution mechanisms to extract low-level features and attention mechanisms to extract complex features, which has shown promising results in various computer vision tasks.

The papers presented in the following sections aim to address the challenges mentioned above. The first paper emphasizes the importance of independent and multicontinental validation sets in developing a reliable and reproducible AI system for prostate cancer detection (Bulten et al., 2022). The second paper focuses on addressing the challenges of image segmentation in whole-slide images (WSI) by proposing an efficient method. The authors demonstrate the effectiveness of ensemble models in achieving accurate image segmentation and provide empirical evidence of their approach (Khened et al., 2021). The third paper proposes a novel architecture, CoTNet, which replaces the convolution layers in ResNet with a block that combines convolution and attention mechanisms. The authors demonstrate that the proposed approach

outperforms ResNet in terms of performance while using fewer parameters. The results suggest the potential of combining convolution and attention mechanisms for improved performance in various computer vision tasks (Li et al., 2022).

2. PANDA CHALLENGE

In recent years, there has been a surge in the prevalence of Artificial Intelligence (AI) competitions which aim to showcase the capabilities, benefits, and limitations of various AI systems (Bándi et al., 2019). Nevertheless, the absence of independent and multinational validations has rendered the outcomes of such competitions unreliable and non-reproducible. Consequently, to promote the creation of sustainable AI systems for Gleason grading utilizing over 10,000 digitalized prostate biopsies, the PANDA challenge was organized. This competition stands out as one of the largest histopathology contests, with the goal of advancing the development of AI technology in the field of prostate cancer diagnosis. Based on the outcomes of the competition, some of the submitted algorithms achieved more than 86% with the uropathologists on external validation sets.

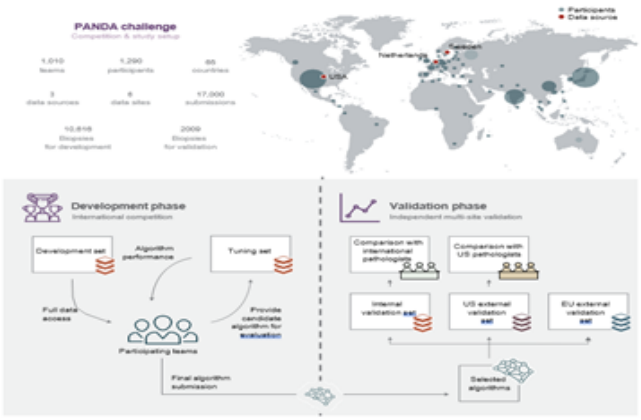


Figure 2. An overview of PANDA challenge

2.1. Gleason Grading

Prostate cancer is a medical condition characterized by the rapid proliferation of cells within the prostate gland. In normal conditions, the cells within the gland tend to grow at a slower pace. There exist five distinct types of prostate cancer, namely, Adenocarcinomas, Small cell carcinomas, Sarcomas, Neuroendocrine tumors, and Transitional cell carcinomas. The most prevalent type among these is Adenocarcinoma. Each of these cancer types is graded based on their level of aggressiveness. This grading system, known as the Gleason Score, is employed to determine the severity of cancer. The Gleason score is a critical diagnostic tool in the management of prostate cancer. This scoring system is used to grade the aggressiveness of prostate cancer, which is a heterogeneous disease that often presents with cells of varying grades. Typically, each patient receives two different grades, one indicating the most prominent case of cancer cells, and the other indicating the second most prominent case of cancer cells. Based on the Gleason score assigned to a biopsy specimen, a patient's biopsy can be assessed as healthy (lower score) or abnormal (higher score).

The Gleason scoring system assigns each of the two most dominant grades of cancer cells a score within the 1 to 5 range. These individual scores are then combined to derive an overall score, which can range from 2 to 10. A Gleason score less than or equal to 6 typically indicates that the tissue sample is non-cancerous or that the patient's condition is not critical (Van Leenders et al., 2020).

2.2. Datasets

The present study focuses on a recent competition that consisted of two primary components. One of the positive aspects of the competition was the diversity of the data, which originated from various institutions in both Europe and the United States. Additionally, an independent blind set of data was provided for validation purposes. By excluding the developers' influence in the selection of the validation sets, the competition organizers ensured a fair and unbiased evaluation process. The dataset consisted of 12,625 whole slide images (WSIs) collected from six sites across Sweden, the Netherlands, and the United States, of which 10,616 and 393 were used for training and tuning, respectively. The validation sets included 545 internal validation data from Europe and 1071 external validation data from the United States. The reference standards for these validation sets were selected from pathologists in the United States and international pathological societies. Figure 2 provides additional details about the origin, number, and division of the

data, as well as the standard references used.

2.3. Scoring Metric

The quadratic kappa score is a statistical measure used to evaluate the level of agreement between two raters or classifiers while taking into account the possibility of chance agreement. It is often employed to assess classification models, especially when there are class imbalances or multiple classes. To calculate the quadratic kappa score, the observed agreement between the two raters or classifiers is compared to the expected agreement due to chance. This method employs the same formula as the kappa statistic but applies a different weighting scheme that places more emphasis on cases where raters or classifiers differ in their assessments. The range of a quadratic kappa score is from -1 to 1. A score of 1 indicates perfect agreement, while a score of 0 indicates no agreement beyond chance. If the score is negative, this indicates that the agreement is even worse than if left to chance. The quadratic kappa score formula is defined as follows:

$$\mathrm{w_{i,j}} = \frac{(\mathrm{i}-\mathrm{j})^2}{(\mathrm{N}-1)^2} \tag{1}$$

$$k = 1 - \frac{\sum_{i,j} w_{i,j}\, O_{i,j}}{\sum_{i,j} w_{i,j}\, E_{i,j}}, \tag{2}$$

in which, w is the difference between actual and predicted values, O is actual outcomes and E is expected outcomes (Gao, 2023).

2.4. Competition Overview

The competition attracted the participation of 1,010 teams hailing from 65 countries, with each team required to submit at least one algorithm. Throughout the contest, all teams were granted access to both the development set and the tuning set. An internal validation set, kept blind, was employed to assess the performance of all algorithms. The top-performing team achieved a quadratic kappa score agreement exceeding 90% with uropathologists. After approximately 33 days, the majority of teams had achieved an 85% agreement, as shown in Figure 3. The aforementioned outcomes lend support to the notion that the majority of algorithms perform comparably, with data management and algorithmic training exerting a more significant influence on performance. In order to compete in external validation sets, 15 teams were selected based on their performance on internal validation sets.

2.5. Methods and Algorithms

The majority of participants in the competition utilized deep learning techniques to devise their algorithms(Hartman et al., 2020). Notably, the top-ranking teams employed patch-based methods, where images were partitioned into smaller patches and subsequently input into the network. The final classifier layer of the network operated on the features obtained by concatenating the resulting features from these patches. This approach is weakly supervised, as it circumvents the need for pixel-level annotation, which is both labor-intensive and expensive, requiring well-trained human resources. Several errors were observed in the labels furnished by pathologists, prompting some participants to engage in label cleaning. To address this issue, incorrect labels were either removed or revised within the training set. This corrective process, commonly known as label denoising, involved identifying instances where the labels deviated significantly from the predicted labels. Subsequently, label denoising was implemented iteratively throughout the model training process. In addition to label cleaning, the top-performing teams in the competition relied heavily on ensemble models. Specifically, ensemble methods were utilized in both the

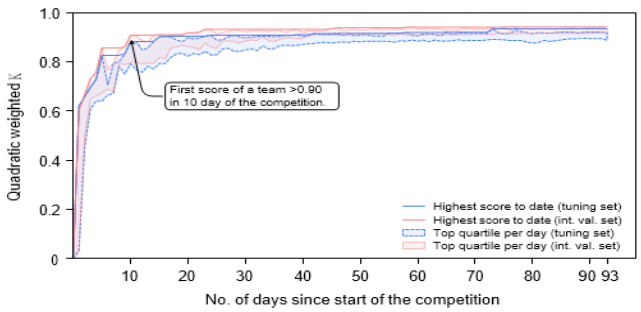


Figure 3. Progression of algorithm performance

preprocessing and classification stages, with the final output results being an average of the output produced by each device and model.

2.6. Classification Performance

Using both internal and external validation sets, the present study evaluates the performance of classification algorithms, as presented in Table 1. The degree of agreement between the algorithms and pathologists was assessed using Kappa statistics at a 95% confidence interval. As indicated by reported sensitivity metrics, representative algorithms overdiagnosed benign cases compared to pathologists' diagnoses, as well as misclassified benign cases more often than they would have been expected based on their performance on an internal validation set. On average, the algorithms missed 1% of cancers in the internal validation set, whereas pathologists missed 1.8%. In the external validation set, the algorithms missed 1.9% of cancers, whereas the pathologists missed 7.3%.

Scores	Internal Validation	EU External Validation	US External Validation
Kappa	0.931	0.868	0.862
Sensitivity	99.7%	97.7%	98.6%
Specificity	92.9%	84.3%	75.2%

Table 1. The external and internal validation results

2.7. Discussion and Limitations

In the PANDA challenge, AI algorithms were developed to detect and categorize prostate cancer. As a way of overcoming previous limitations, the challenge promoted reproducibility and indiscriminate validation of algorithms by a wide range of groups. In the competition, algorithms were produced that were capable of detecting and grading tumors, meeting expert reference standards comparable to those used by pathologists. However, the algorithms tended to assign higher grades than pathologists did, indicating that AI-supported general pathologists could achieve better agreement with uropathologists.. Based on the short lead time of top-performing solutions by various teams, the publication of such datasets could facilitate the development of high-performance AI algorithms. There are a number of limitations to this study, including the fact that only 15 teams were included in the validation phase from a pool of 1,010, which may limit the generalizability of the findings. The algorithm validation was also restricted to single biopsies, while pathologists typically examine multiple biopsies per patient. Moreover, the study graded only one type of prostate cancer. When algorithms are compared against reference standards established by different panels of pathologists, there is a risk of bias, since they may have learned grading habits that are not applicable to other populations. The study was mainly conducted in white-dominated countries, and certain demographic information was not available.

3. A DEEP LEARNING FRAMEWORK FOR WHOLE-SLIDE IMAGE SEGMENTATION

The use of deep learning approaches presents a number of technical difficulties in the field of histopathology tissue analysis. These issues include the extensive size of WSI data, variations in the images themselves, and the complexity of features. To address these challenges, the authors present a comprehensive deep-learning framework that is specifically designed for this type of analysis. their framework is composed of several individual techniques that are applied in a sequence throughout the preprocessing-training-inference pipeline. By combining these techniques, the analysis becomes more efficient and generalizable. These techniques include an ensemble segmentation model, dividing WSI into smaller, overlapping patches, handling class imbalance problems, efficient inference methods, and uncertainty estimation. Their ensemble consists of three deep neural networks: DenseNet-121, Inception-ResNet-V2, and DeeplabV3Plus (Chen et al., 2018), each trained end-to-end for every task. Their framework has been demonstrated to be effective and generalizable through the evaluation of breast cancer metastases (CAMELYON), colon cancer (DigestPath), and liver cancer (PAIP).

3.1. Datasets

The datasets employed to assess the proposed methodology encompass CAMELYON (Litjens et al., 2018), which consists of 1399 whole-slide images (WSIs), PAIP (Kim et al., 2019), comprising 90 WSI slides, and DigestPath (Da et al., 2022), containing 872 tissue images. Pertinent details regarding the dimensions of the training and testing datasets, as well as the resolution and size of the images, can be found in Table 2. The CAMELYON16 data set comprises two distinct classes of slide images, namely metastases, which correspond to cancerous tissues, and negative, which denotes the absence of cancerous tissue. In CAMELYON17, the slide-level labels consist of negative, micro, macro, and ICT, based on the size of the tumor. Notably, some portions of the dataset possess pixel-level annotations, while others are annotated at the slide level. The PAIP images consist of pixel-level annotation, of the viable tumor and whole tumor regions. The DigestPath dataset comprises tissue samples gathered during the examination of colonoscopy pathology with the aim of detecting the presence of early-stage colon tumor cells. A single whole-slide image (WSI) in this dataset contains at least ten tissue sections, each of which is evaluated during colonoscopy pathology review.

To facilitate analysis, the challenge organizers selected one or two tissue sections from each WSI image and made available the corresponding lesion annotations, which were provided by pathologists and saved in jpg format alongside the tissue section images. Some of the images were annotated based on just slide level, and some others pixel-level.

Dataset	Train	Test	Image Size	Pixel
CAMELYON16	270	129	100,000 ×100,000	0.25
CAMELYON17	500	500	100,000 ×100,000	0.25
DigestPath	660	212	5000 × 5000	0.25
PAIP	50	40	50,000 × 50,000	0.50

Table 2. Characteristics of datasets used for experiments

3.2. Proposed Framework

The authors of the study employed a general strategy that consisted of ensemble network architectures, training strategies, and segmentation inference methods. Additionally, the strategy included methods for performing secondary histopathology analysis, which is a common approach to further evaluate the characteristics of a tumor after surgical removal and examination by a pathologist. The overall process is visually represented in Figure 4.

3.3. Network Architecture

To segment the tumor region from patches extracted from WSI images, an ensemble Fully Convolutional Network (FCN) architecture is employed. Segmentation networks based on FCNs usually consist of encoders, decoders, and pixel-wise classification layers. In order to produce a low-resolution feature map, the encoder network uses a combination of convolution and pooling operations. In the decoding network, the low-resolution feature map is then upsampled and convoluted back to the original resolution via upsampling and convolution operations. As a result, the WSI images can be segmented accurately using FCN ensembles. Three encoders and decoders make up the ensemble architecture. Different encoder-decider combinations are in the following architectures:

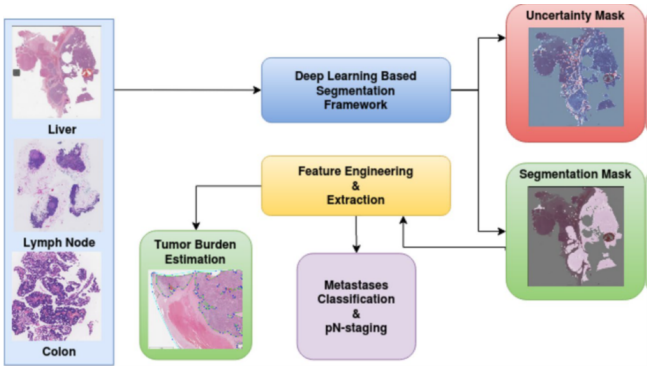


Figure 4. The process of the segmentation

- U-Net (Ronneberger et al., 2015) architecture with DenseNet-121 as the pre-trained encoder on ImageNet, and the decoder was a combination of bi-linear upsampling modules followed by convolutional layers. Bilinear

upsampling is a commonly used technique in deep learning models, especially in image classification and object detection tasks, as it can improve the accuracy of the model's predictions by increasing the resolution of an image before inputting it into the model. Bilinear upsampling works by inserting new pixels between the existing pixels in an image and each new pixel is calculated using a weighted average of the four nearest pixels in the original image, with weights determined by the distance from the surrounding pixels. Additionally, the authors applied skip connection, where extracted features from the encoder were concatenated with extracted features from the decoder. This technique helps to maintain the resolution of the image during the encoding and decoding processes and preserves important spatial information for more accurate segmentation results.

- A combination of U-Net and Inception-ResNet-V2 as the pre-trained encoder on ImageNet. The Inception-ResNet-V2 architecture, also referred to as Inception-v4, is a convolutional neural network that merges the characteristics of two previously developed models: Inception and ResNet. The integration of both architectures in Inception-ResNet-V2 involves using multi-scale convnet blocks from the Inception network to decrease the number of parameters while encoding a considerable amount of information. These blocks comprise various convolutions with diverse kernel sizes, accompanied by pooling layers that reduce the spatial dimensions of feature maps. This enables the network to extract features at different scales, capturing fine and coarse details in the input image.
- Deeplab V3Plus with Xception (Chollet, 2017)as the encoder, which is pretrained on PASCAL VOC (Everingham et al., 2010). To achieve multi-scale context, DeepLabV3Plus employs atrous convolutions with varying rates. These dilated convolutions enhance the network's receptive field without increasing the model's number of parameters. By applying atrous convolutions with different rates, DeepLabV3Plus captures contextual information at various scales, enabling accurate semantic segmentation. The DeepLabV3Plus architecture also incorporates a feature pyramid network that transfers low-level features from the encoder to the decoder. This is achieved using skip connections that concatenate feature maps from different layers in the encoder with corresponding feature maps in the decoder, preserving crucial spatial information and improving the segmentation accuracy.

3.4. Training Pipeline

The training pipeline is a combination of tissue mask generation, patch extraction, and training the models patch-wise. the pipeline can be found in Figure 5.

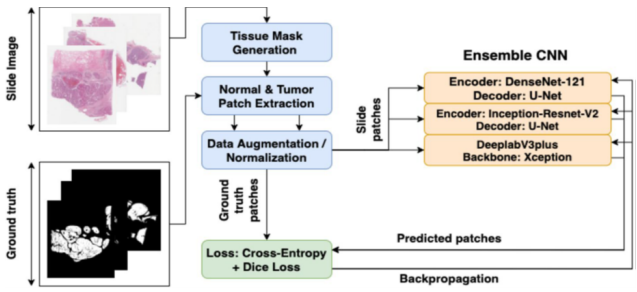


Figure 5. An overview of the training pipeline

3.4.1. Tissue Mask Generation

WSI image segmentation was performed by separating the tissue region from the background glass region. By segmenting, unnecessary data is not computed on non-tissue regions. WSI images with low resolution are used since an approximate boundary of the tissue region is enough. This step converts the RGB color space from the low-resolution image to the HSV color space. A binary image is created based on the saturation component, which is thresholded using Otsu's adaptive thresholding method. Using binary morphological operations, small tissue regions and tissue boundaries are accurately extracted from the image.

3.4.2. Patch Coordinate Extraction

After generating the tissue mask, the next step involved randomly extracting patches of the image to create the training dataset. To prevent class imbalance and ensure proper training, an equal number of tumorous and non-tumorous patches were extracted from the tissue mask. A patch was classified as tumorous if at least one pixel inside the patch was labeled as a tumor. Patch extraction was done in higher resolution.

3.4.3. Data Augmentation

Data augmentation methods were used to increase the number of data points to improve generalization across staining and acquisition protocols. Various augmentations were applied, including "horizontal or vertical flipping", "90-degree rotations", "Gaussian blurring", and color augmentation. Random changes were made to brightness, contrast, hue, and saturation during color augmentation. Using random coordinate perturbation, patches extracted from images at different epochs were more diverse. An offset of 128 pixels is performed before the patch is extracted from the WSI image. A normalization process was performed after augmentation.

3.4.4. Loss Function

The images of WSI showed that the tumor regions were only a tiny fraction of the total pixels, creating a class imbalance. To overcome this challenge, the network was trained utilizing a hybrid loss function that minimizes this issue. There are two types of loss functions in the hybrid loss function: cross-entropy loss and dice loss. In image segmentation tasks, cross-entropy loss is a loss function that calculates the difference between predicted probabilities and actual labels. Meanwhile, dice loss measures the overlap between the predicted and true segmentation maps, which helps evaluate segmentation quality. In order to determine the dice loss, the predicted posterior probability map is used along with the ground truth binary image. Using the predicted posterior probability map, the network assigns a probability value to each pixel indicating whether the pixel belongs to the tumor region. Ground truth binary images label pixels as either belonging to the tumor region or not. Dice loss is defined as the difference between the predicted segmentation map and the true segmentation map, which is a measure of overlap. Through the integration of cross-entropy loss and dice loss, a hybrid loss function is created that optimizes both pixel-wise classification accuracy and true segmentation similarity. Studies have shown that this method improves segmentation performance. The formula for the hybrid loss function is:

$$DL = 1 - \frac{2\sum_i^N p_i g_i}{\sum_i^N p_i^2 + \sum_i^N g_i^2} \tag{3}$$

$$CL = \sum_i^N \left(g_i log(p_i) + (1 - g_i) log(1 - p_i) \right) \tag{4}$$

$$Loss = \alpha * CL + \beta * DL_{BG} + \gamma * DL_{FG} \tag{5}$$

In these formulas, p_i and g_i are a pair wise pixel value predicted posterior and ground truth. N is the total number of pixels, DL represent the dice loss and CL refers to cross-entropy loss. DL_{FG} and DL_{BG} are foreground pixels match to the tumor regions and background pixels correspond to the non- tumor regions. The α, β, γ are assigned in a way that the cross-entropy loss and the dice loss have the equal weights.

3.4.5. Training

The three neural networks were trained using distinct cross-validation folds, with the encoder component of each network initialized with pre-trained weights. For two networks that employed DenseNet-121 and Inception-ResNet-V2 as encoders, the weights of the pre-trained model were fixed during the initial two epochs, with only the decoder weights being trained. Subsequently, both encoder and decoder weights were trained for the remaining epochs. A decayed learning rate based on the number of epochs was implemented to capture the gradual convergence of the model. Once the validation loss began to increase, the training process was terminated.

3.5. Inference Pipeline

To facilitate patch extraction, tissue regions were segmented from WSIs in the preprocessing step. To generate the uniform patch-coordinate sampling grid, a lower resolution grid was generated, which was then scaled to match the highest resolution WSIs. Scaled coordinate points were used as centers for extracting high-resolution image patches from the WSI image. The stitching of segmented patches caused boundary artifacts due to the smaller patch sizes that could not capture the context of a larger neighborhood region. As a solution to these issues, the inference was made on overlapping patches with large patch sizes,

with average prediction probabilities calculated at regions that overlapped. It was found to be optimal to overlap consecutive neighboring patches by 50% for both accuracy and computation efficiency. Furthermore, an increase in patch size by a factor of four during inference increased the quality of generated heatmaps compared to those generated during training.

3.6. PN-staging

PN-staging was applied for CAMELYON17 dataset. PN-staging allows doctors to more accurately assess the extent of cancer and its response to treatment, which can help guide further treatment decisions. Figure 6 shows the pipeline of PN-staging.

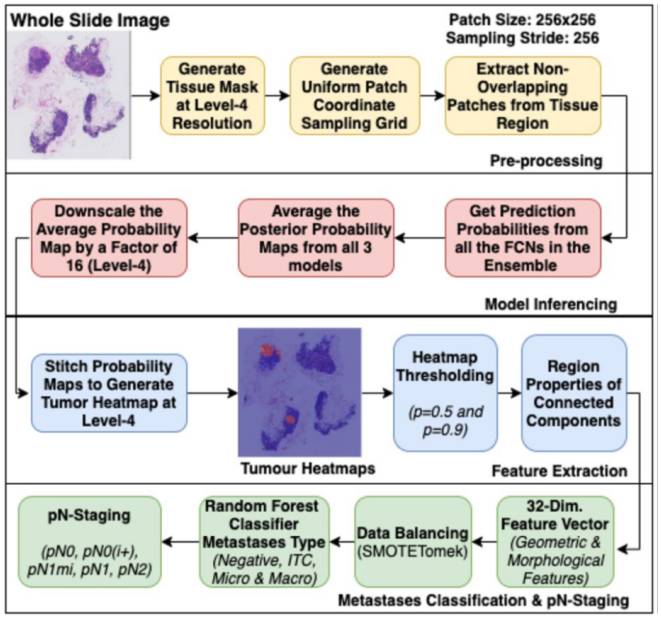


Figure 6. An overview of the PN-staging

The steps for PN-staging are as follows:

- Preprocessing: This step involved detecting tissue regions in whole slide images (WSI) for patch extraction.
- Heatmap generation: The extracted patches were then used to generate down-scaled tumor probability heatmaps via an inference pipeline.
- Feature extraction: feature extraction was carried out by binarizing the heatmaps at 0.5 and 0.9 probabilities, extracting connected components, and measuring region properties using the scikit-image (Van Der Walt et al., 2014) library. A total of thirty-two geometric and morphological features were computed from the probable metastases regions. Table 3 shows the results.

No.	Feature Description	Threshold
1	Largest tumor region's major axis length	0.9, 0.5
2	Largest tumor region's area	0.5
3	Ratio of tumor region to tissue region	0.9
4	Count of none-zero pixel	0.9
5	Tumor regions area	0.9
6	Tumor regions perimeter	0.9
7	Tumor regions eccentricity	0.9
8	Tumor regions extent	0.9
9	Tumor regions solidity	0.9
10	Mean of all regions mean confidence probability	0.9
11	Number of connected regions	0.9

Table 3. Extracted features for predicting lymph node metastasis type

- Data balancing: As a means of addressing class imbalance, oversampling of SMOTE algorithm (Chawla et al., 2002) was employed, although this method can introduce noise. To eliminate noisy samples, under-sampling techniques such as Tomek's link or nearest neighbors were employed. In this study, SMOTETomek (Batista et al., 2004), a combination of SMOTE and Tomek's link, was used to balance the training data.
- Classification**:** Assigning the PN-stage to the patient was determined based on the available lymph node WSI images, taking into account the type of metastases each patient had (Table 4). The extracted features were used to train an ensemble of Random Forest classifiers to predict metastases type.

3.7. Tumor Burden Estimation

Tumor burden estimation is a critical step in the analysis of PAIP datasets related to liver tissues. It is a process that involves the measurement of the amount of cancerous tissue present in a patient's liver cancer

Category	Size
Isolated Tumor Cells	Single tumor cells or a cluster of tumor cells $\leq 0.2\ mm$ or less than $< 200\ cells$
Micro Metastasis	$\geq 0.2\ mm$ and/or $> 200\ cells$ and $\leq 2\ mm$
Macro Metastasis	$\geq 2\ mm$

Table 4. The assigned metastasis type based on the tumor size

tissue. In order to carry out this process, it is necessary to segment the viable tumor and whole tumor regions within the liver tissue. The viable tumor region specifically refers to the portion of the tumor that is actively growing and dividing, and this region is identified through the use of a deep learning-based segmentation network. While this network proved to be effective in identifying the viable tumor region, training it for the whole tumor region did not yield optimal results. As a result, a heuristic approach was employed to approximate the whole tumor region from the viable tumor region.

The steps in tumor burden estimation are as follows:

- Segment the viable tumor region: This involves using the algorithm proposed in the "Inference pipeline" section to segment the portion of the tumor that is actively growing and dividing. This step involves the use of deep learning-based segmentation networks.
- Apply morphological operations: The segmentation process may result in false positives or small holes in the prediction. Applying morphological operations (e.g., erosion, dilation) can help to remove these issues.
- Find the smallest convex hull: This involves identifying the smallest convex hull that contains the entire viable tumor region. The convex hull is the smallest convex polygon that can enclose all the points in a given region.
- Approximate the whole tumor region: As training the same segmentation network for the whole tumor region resulted in sub-optimal outcomes, a heuristic approach was adopted to approximate the whole tumor region. This involves intersecting the convex hull with the tissue mask to identify the whole tumor region.
- Calculate the tumor burden: This step involves taking the ratio between the area of the viable tumor region and the area of the whole tumor region to calculate the tumor burden. This provides a quantitative measure of the amount of cancerous tissue present in the liver tissue.

3.8. Uncertainty Analysis

Uncertainty analysis refers to the process of estimating and quantifying the level of uncertainty in the predictions made by machine learning models. In the context of medical diagnosis, uncertainty analysis can be used to identify cases where the model's predictions are unclear or uncertain, which can then be flagged for further review by human experts. There are two main sources of uncertainty in machine learning models: aleatoric uncertainty and epistemic uncertainty. Aleatoric uncertainty arises from the inherent variability in the data generation process and can be estimated using techniques such as test time augmentations. Epistemic uncertainty (Kendall & Gal, 2017), on the other hand, arises from the model architecture and parameters and can be estimated using techniques such as test time Bayesian dropout. Bayesian dropout is a regularization technique that can be applied to deep neural networks during training to prevent overfitting and improve generalization. It works by randomly dropping out neurons during training with a certain probability, which forces the network to learn more robust features that are not dependent on any particular set of neurons. During test time, Bayesian dropout can be used to estimate the model's uncertainty by running multiple forwards passes through the network with dropout applied and computing the variance of the predictions. This can be interpreted as a measure of how much the model's predictions vary depending on which neurons are dropped out, and hence how uncertain the model is about its predictions.

In the proposed pipeline, aleatoric uncertainty for each model was estimated using test time augmentations (TTA), which involves applying various transformations to the input images at test time to generate multiple predictions for each image. The aleatoric uncertainty for a given image was then estimated as the variance of the predictions across the different augmentations, using:

$$var_{al}(x, \varphi_i) \approx E_{t\sim TTA}[(\varphi_i(x\,|w, t) - E_{t\sim TTA}[\varphi_i(x|w, t)])^2] \quad (6)$$

Here, x represents the input image, φ_i represents the machine learning model, and w and t represent the weights and dropout probabilities, respectively, for the model. The term $\varphi_i(x|w, t)$ represents the output of the neural network with weights w and with dropout applied for input x. $E_{t\sim TTA}$ denotes the expected value across all possible augmentations, and the formula calculates the variance of the predicted probabilities across the different augmentations, which is a measure of how much the predictions vary depending on the specific data augmentation applied. In the context of machine learning, epistemic uncertainty refers to uncertainty that arises from the model itself, such as the model architecture, hyperparameters, and weights. In order to estimate this type of uncertainty, a diverse set of model architectures can be used. Equation (7) shows the formula used for epistemic uncertainty.

$$var_{ep}\big(p(y|x, w)\big) \approx E_{\varphi\sim\{\varphi_i\}}\left[\big(\varphi(x|w) - E_{\varphi\sim\{\varphi_i\}}[\varphi(x|w)]\big)^2\right] \quad (7)$$

Where $p(y|x, w)$ is the likelihood distribution of the probabilistic model, which generates outputs y for given inputs x for some parameter settings w. The notation φ_i indicates the output of a specific trained model i for input x and with parameters w.

3.9. Results

Performance evaluation on CAMELYON17 is shown in Table 5. In the CAMELYON17 testing dataset comprising 500 cases, an ensemble approach was used to combine the predictions made by four trained Random Forest classifiers. This ensemble model, called RF-Ensemble, utilized the majority voting principle to make the final prediction. In the event of a tie, the higher metastases category was chosen. The results of the proposed ensemble approach were compared with other published approaches on the same dataset, and it achieved a Cohen's kappa score of 0.91.

In Khened et al. (2021), 212 samples from DigestPath-2019 were used to test the proposed approach against other methods. Among the methods compared, the proposed method achieved a Dice score of 0.78. This score measures the similarity between two sets of data, such as segmenting an image into foregrounds and backgrounds. An overlap of 1 indicates a perfect match between two sets, and a match of 0 indicates there is no overlap at all. The dice score is calculated by dividing twice the overlap between two sets by the sum of their sizes. A higher Dice score indicates better segmentation accuracy (Table 6).

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

Table 5. Kappa scores on CAMELYON17.

Table 7 provides a comparison between the proposed approach and other methods on the PAIP-2019 dataset. A total of two tasks were included in this dataset. In Task 1, the Jaccard index was used to evaluate the performance of liver cancer segmentation, and in Task 2, the combination of absolute accuracy and the Jaccard index was used to estimate viable tumor burden for each case in the test set. Participants in Task 1 used deep learning methods, but with different CNN architectures. For Task 2, participants used deep learning-based methods for segmenting the whole tumor. Comparatively to deep learning-based methods, the proposed convex hull-based method performed well. In image segmentation, algorithm performance is commonly evaluated by comparing predicted results with ground truth results.

Teams	Dice
Kuanguang	0.807
Zju_realdoctor	0.792
TIA_Lab	0.787
propose	0.782

Table 6. Dice scores on DigestPath.

Teams	Task 1	Task 2
FNLCR	0.789	0.752
Sichuan University	0.777	NA
Proposed	0.750	0.634
Alibab	0.672	0.620
Sejong University	0.665	0.630

Table 7. Top five entries of PAIP-2019.

3.10. Discussion and Conclusion

For the segmentation and downstream analysis of WSI images, the article presents an automated end-to-end deep learning framework. By divide-and-conquer strategy, large images are divided into smaller patches, which are then classified at the pixel-level using an ensemble of FCNs. Based on publicly available histopathology image analysis challenges, the proposed framework achieved state-of-the-art results. Among the advantages of the proposed method is its superior segmentation performance using an ensemble approach, as well as its ability to generate uncertainty maps for better pathologists' interpretation.

4. CONTEXTUAL TRANSFORMER NETWORKS FOR VISUAL RECOGNITION

The Transformer architecture with self-attention has been a significant breakthrough not only in natural language processing but also in computer vision. However, existing Transformer designs rely on self-attention directly over 2D feature maps, where queries and keys are specified at each spatial location and ignore the contextual relationships between neighboring keys. To address this limitation, this paper proposes a Transformer-based method called Contextual Transformer (CoT), which utilizes contextual information among input keys to enhance the results. Using a three-by-three convolution, the CoT block encodes keys in static form. Two 1x1 convolutions are used to learn the dynamic multi-head attention matrix by concatenating encoded keys with input queries. For dynamic contextual representations of inputs, the learned attention matrix is multiplied by input values.

The Transformer architecture performs better with a CoT block that outputs a combination of static and dynamic contextual representations. Using Contextual Transformer Networks (CoTNet), every 3x3 convolution in ResNet architectures is replaced by a CoT block. It enables us to create a powerful network that makes full use of context information between input keys. Different experiments have been conducted to assess the effectiveness of the proposed approach, including image recognition, object detection, instance segmentation, and semantic segmentation. The comparison between the conventional self-attention and the contextual transformer block can be found in Figure 7 and Table 8. In Figure 7, (a) shows a conventional transformer block and (b) shows a contextual transformer block. They demonstrated the power of CoTNet by comparing it with several state-of-the-art backbones. When comparing CoTNet to ResNeSt (101 layers), it achieves an absolute reduction of 0.9% in top-1 error rate. The performance of CoTNet on COCO (Lin et al., 2014) for object detection and instance segmentation is superior to ResNeSt (Zhang et al., 2020). As well, CoTNet performed 1.8% better than DeiT-B (Touvron et al., 2021) for semantic segmentation on ADE20K (Zhou et al., 2019).

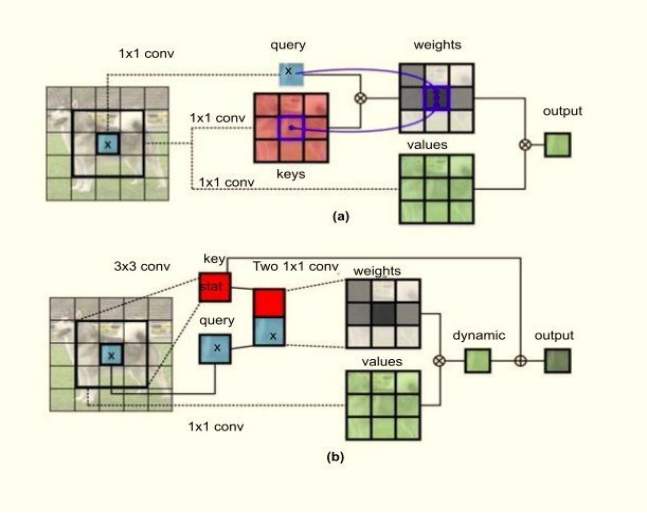


Figure 7. Comparison of transformer blocks

4.1. Transformer

Transformers are a popular type of neural network architecture used in natural language processing (NLP) and other sequential data

Algorithm	ResNet-50	CoTNet-50	ResNeXt-50	CoTNeXt-50
Parameters	25.56×10^6	22.21×10^6	25.03×10^6	30.05×10^6
FLOPs	4.12×10^9	3.28×10^9	4.27×10^9	4.33×10^9

Table 8. Comparison of FLOPs and Parameters of two contextual networks and their corresponding backbone.

applications. These networks utilize self-attention to focus on different parts of input sequences at different times. A transformer architecture typically consists of an encoder and decoder, each containing multiple layers of self-attention and feedforward neural networks.

- Encoder: The encoder takes an input sequence and transforms it into a sequence of vectors through embedding. A self-attention layer assigns attention weights to each token in the sequence, allowing the network to determine the importance of each token in generating the output. The output of the self-attention layer then passes through a feedforward neural network, which nonlinearly transforms each token's vector representation, before being normalized across the entire sequence.
- Decoder: Similarly, the decoder generates output tokens based on the encoder's output. The input sequence to the decoder is also embedded into a sequence of vectors and passed through a mask attention layer to prevent cheating during training. Attention weights are calculated by passing the encoder output and the previous decoder layer's output through an attention layer. The output then passes through feedforward neural networks and layer normalization before generating a probability distribution over the output vocabulary. The token with the highest probability is selected as the next output token.

Transformers allow for parallel computations and can handle input sequences of varying lengths. They also capture long-range dependencies using self-attention, making them suitable for various NLP tasks. Figure 8 shows a general architecture of the transformers.

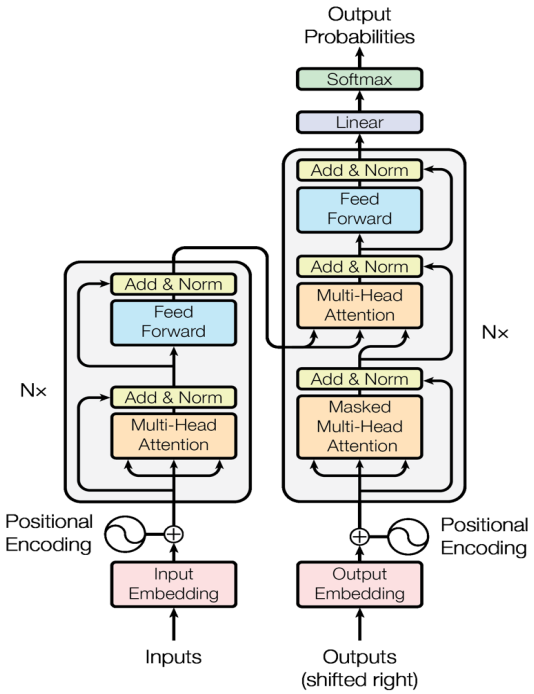


Figure 8. The transformer block.

4.2. Self-attention

As a result of an attention mechanism, a query and a group of key-value pairs are mapped into an output, where each pair is displayed as a vector. Each value is weighted according to its relationship with its corresponding key based on a compatibility function. Their attention mechanism is called "Scaled Dot-Product Attention," and is defined as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (8)$$

and is shown in Figure 9. By calculating the dot products of the query and all keys, dividing the results by d_k square root, and applying a softmax function, the weights for those keys are obtained. The results of several queries are written into a matrix Q, with keys and values going into matrices K and V.

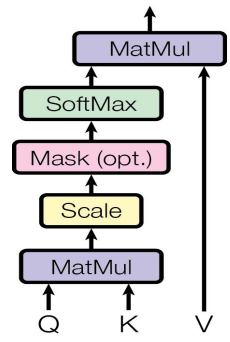


Figure 9. The attention mechanism.

4.3. Self-attention in Computer Vision

In response to Transformer's outstanding performance in various NLP tasks, researchers have begun exploring its application to vision tasks. The purpose of self-attention mechanisms in NLP sequence modeling was originally to capture long-range dependencies. Self-attention in the vision domain can involve applying feature vectors to different areas of an image. Local self-attention within a local patch has been found to be an effective alternative to global self-attention over an entire feature map (Hu et al., 2019). An algorithm for self-supervised representation learning was developed based on transforming raw images into 1D sequences (Chen et al., 2020). In order to detect objects and recognize

images, Pure Transformer uses sequences of local features or image patches. The 3×3 convolutions in a ResNet were replaced with layers of self-attention in (Srinivas et al., 2021).

According to some recent works, the input image is divided into several patches as "visual sentences" and then into sub-patches as "visual words," in contrast to ViT, which only divides it into patches. Swin Transformer further improves ViT by merging image patches in deeper layers, which has linear computation complexity as image size increases (Liu et al., 2021). For local and global attention interleaving with high throughput, Twins-PCPVT and Twins-SVT (Twins transformers) have been proposed (Chu et al., 2021). There has been developed a technique known as cross-covariance attention (XCA), which makes use of tokens (words or image patches) instead of feature channels, which produces a linear approach to self-attention (El-Nouby et al., 2021).

4.4. Multi-head Attention in Vision Backbone

Multi-head attention in the vision backbone uses embedding matrices, to transform 2D feature map X into queries Q, keys K, and values V. This is done as a 1x1 convolution in space. After that, a local matrix multiplication operation is performed between keys K and queries Q to obtain the local relation matrix R. R represents all local query-key relation maps for each head. 2D relative position embeddings are integrated into the local relation matrix R to provide position information about each grid. To achieve attention matrix A, the enhanced local relation matrix is normalized using the Softmax operation along the channel dimension. The final feature map is derived by aggregating all values within each grid with the learned local attention matrix. Finally, the output Y is the concatenation of aggregated feature maps for all heads. Equations and methods for Multi-head attention in vision backbone are:

$$R = K \circledast Q \tag{9}$$

$$\hat{R} = R + P \circledast Q \tag{10}$$

$$Y = V \circledast A\,, \tag{11}$$

A block diagram for Multi-head attention is given in Figure 10, where $R \in \mathbb{R}^{H\times W\times(K\times K\times C_h)}$ and $P \in \mathbb{R}^{K\times K\times C_h}$, C_h is the head number, and the operator $\circledast$ depicts local matrix multiplication.

4.5. Proposed Method

Present methods rely primarily on conventional self-attention and ignore the explicit modeling of relationships between adjacent keys. Contextual Transformer provides both context discovery among keys and self-attention learning over the feature map in a single architecture. Contextual information is incorporated into input keys to enhance self-attention learning, resulting in improved representational abilities for deep networks. Contextual Transformer Networks derived from ResNet and ResNeXt (Xie et al., 2017) are subsequently used to replace 3x3 convolutions throughout the deep architecture.

4.5.1. Contextual Transformer Block

Self-attention mechanisms typically allow feature interactions to occur across various spatial locations depending on the inputs. This mechanism, however, does not explore the contexts between pairwise query-key relations, leading to restricted self-attention learning over 2D feature maps. Contextual Transformer (CoT) blocks have been created to address this limitation. In the CoT block, contextual information mining and self-attention learning are linked into a single architecture to enhance the representativeness of the aggregated feature map. By applying k × k group convolutions over

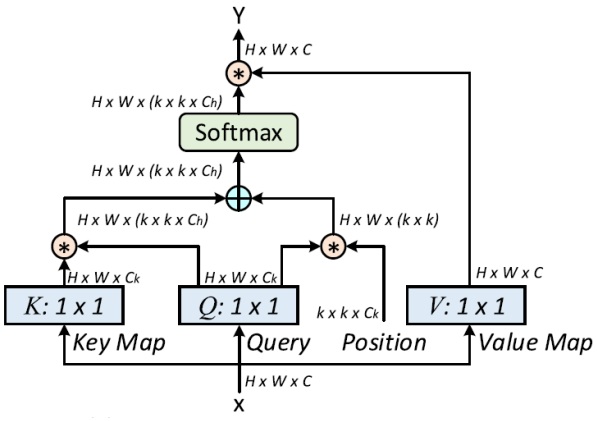


Figure 10. The multi-head attention block.

all the neighbor keys within the k × k grid, CoT contextualizes each key representation. Learning contextualized keys reflects static context information among local neighbors. In order to achieve the attention matrix, they performed two consecutive 1×1 convolutions using the contextualized key feature and the query feature. Based on the contextualized attention matrix, the attended feature map is calculated. CoT is a block that fuses static and dynamic contexts. Through an attention mechanism, it adaptively aggregates the two contexts into a single final output. Equations are:

$$A = [K^1, Q]W_\theta W_\delta \quad (12)$$

$$K^2 = V \circledast A\,. \quad (13)$$

Figure 11 shows a contextual transformer block in a systematic way.

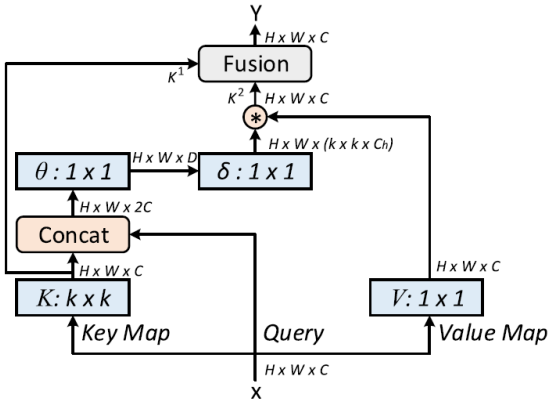


Figure 11. The contextual transformer block.

4.5.2. Contextual Transformer Network

By using the CoT block instead of each convolution layer in ResNet, they were able to create a high-performing network without dramatically increasing the parameter budget. The Contextual Transformer Networks (CoTNet) are based on ResNet-50 and ResNeXt-50 backbones and show the two different constructions, respectively, of CoTNet-50 and CoTNeXt-50. With CoTNet-50, all the convolutions in stages res2, res3, res4, and res5 are replaced with CoT blocks. CoTNet-50 has a similar number of parameters and FLOPs to ResNet-50 due to the similar computational nature of CoT blocks. CoTNeXt-50 is built by replacing all the three convolution kernels in ResNeXt-50 with CoTs. This kernel depth decreases significantly with an increase in groups, however. Due to this reduction in depth, group convolutions in ResNeXt-50 are computationally cheaper by a factor of C (Table 8).

4.6. Experiments

CoTNet has been evaluated over some computer vision (CV) applications in order to prove its effectiveness. Among the applications are image recognition, object detection, instance segmentation, and semantic segmentation. As a first step, CoTNet was trained from scratch using the ImageNet benchmark (Deng et al., 2009). Afterward, the trained network has been evaluated for object detection and instance segmentation on COCO datasets, and for semantic segmentation on ADE20K datasets. This report exclusively focuses on the evaluation of time versus accuracy trade-offs achieved through default and advanced training methods in the context of image recognition. Additionally, the report investigates the impact of various CoT block designs on the performance of CoTNet-50. While the outcomes presented here are limited to image recognition, readers interested in the results pertaining to semantic segmentation, instance segmentation, and object detection are directed to (Li et al., 2022).

4.6.1. Image Recognition

A dataset called ImageNet (Deng et al., 2009) is used in the image recognition task, which is comprised of 1.28 million training images and 50,000 validation images from 1,000 classes. The validation set is evaluated by reporting the top-1 and top-5 accuracy. Experimental setups include two different kinds of training setups, namely the default training and advanced training. Typical training for networks such as ResNet, ResNeXt, and SENet (Hu et al., 2018) involves training networks for approximately 100 epochs. For the dataset, the augmentation techniques are done, and all hyperparameters are set as per the original implementations. Using backpropagation, the CoTNet is also trained end-to-end using SGD with momentum 0.9, label smoothing 0.1, and batch size equal to 512. Cosine schedules are used to decay the learning rate in the first five epochs. CoTNets are trained to 350 epochs, with RandAugment data augmenting, and dropout and DropConnect regularization.

4.6.2. Performance

A number of recent vision backbones were compared with two different training settings (default and advanced) using the ImageNet dataset. CoTNet and CoTNeXt have been built using two different depths: 50 layers and 101 layers, resulting in CoTNet-50/101 and CoTNeXt-50/101. The advanced training setup featured an upgraded version of CoTNet, SE-CoTNetD-101, in which the 3x3 convolutions in the res4 and res5 stages were replaced by CoT blocks under the SE-ResNetD-50 backbone (He et al., 2019). Based on the default training results, CoTNet-50/101 and CoTNeXt-50/101 outperformed current vision backbones consistently across both top-1 and top-5 accuracy levels, including both ConvNets (e.g., ResNet-50/101 and ResNeXt-50/101) and attention-based models (such as Stand-Alone and AA-ResNet-50/101). In this study, we have demonstrated an effective way of enhancing visual recognition by combining context mining and self-attention learning into a single architecture.

4.6.3. Inference Time Versus Accuracy

Using both inference time and top-1 accuracy, the CoTNet models were evaluated on image recognition tasks. Both Figure 12 and Figure 13 illustrate the inference time-accuracy curves of CoTNet and state-of-the-art vision backbones for default and advanced training setups. The CoTNet models demonstrated better top-1 accuracy at lower inference times than other vision backbones for both training setups when compared with other vision backbones. Compared to Efficient-Net-B6, SE-CoTNetD-152 (320) achieved 2.75x faster inference speeds while achieving 0.6% higher top-1 accuracy.

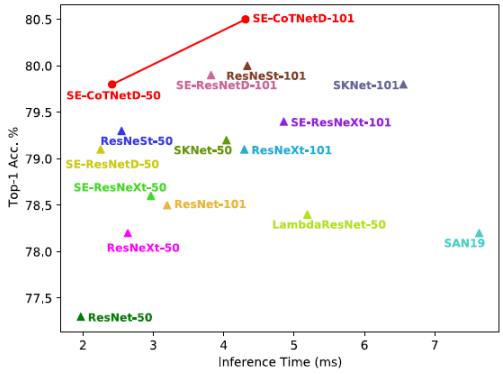


Figure 12. Time versus accuracy-default training

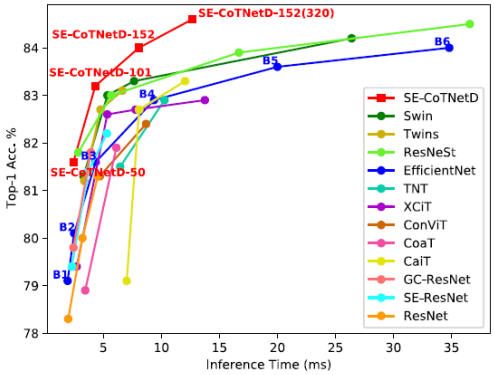


Figure 13. Time versus accuracy-advanced training

4.6.4. Effect of Different Configurations

Detailed analyses of each CoT block design are performed in this section in order to assess their impact on the overall performance of CoTNet-50. Initially, the static context is located among the keys through a convolution of three, and then the dynamic context is formed by concatenating the query and contextualized key. CoT blocks combine the static and dynamic contexts to produce the final outputs. Linear Fusion and Concatenate are two variations of CoT blocks that directly sum or concatenate two contexts. Table 9 outlines the performances of various ways of exploring contextual information in CoTNet-50 backbone. According to the results, 77.1% of the top-1 accuracy was achieved using the static context alone. In contrast, the exploitation of dynamic context shows improved performance. By combining static and dynamic contexts linearly, 78.7% of the gains occur, indicating their complementarity. Additionally, Concatenate is comparable to Linear Fusion in terms of performance (Table 9).

4.6.5. Effect of Replacement Settings

ResNet-50 backbone performance is analyzed in relationship to the number of stages replaced with CoT blocks in order to find a better balance between speed and accuracy. As shown in Table 10, replacing more stages with CoT blocks generally improves performance while decreasing parameter numbers and FLOPs

slightly. Performance boosts are most notable in the last two stages (res4 and res5) after CoT blocks are replaced. Moreover, the additional replacement of CoT blocks in the first stages (res2 and res3) only leads to marginal performance improvements and $1.34x$ inference time increases, as shown in Table 10. As the results show, SE-CoTNetD-50 achieved better performance with a negligible decrease in throughput compared to SE-ResNetD-50.

Algorithm	Params	FLOPs	Top-1 Accuracy	Top-5 Accuracy
Static Context	17.1 M	2.7	77.1	93.5
Dynamic Context	20.3 M	3.3	78.5	94.1
Linear Fusion	20.3 M	3.3	78.7	94.2
Concatenate	22.8 M	3.7	78.9	94.3
CoT	22.2 M	3.3	79.2	94.5

Table 9. The results of different configurations of CoT block

4.7. Conclusion

The authors introduced a novel architecture called the Contextual Transformer (CoT) block that combines contextual information with self-attention learning to improve visual representation. The CoT block captures static context among input keys and uses it to trigger self-attention for mining dynamic context. This approach replaces standard convolutions in ResNet architectures and results in better performance without increasing parameters. The authors construct CoTNet by replacing 3x3 convolutions in ResNet architectures, which validates their proposal and analysis. Extensive experiments on COCO and ADE20K datasets demonstrate the generalization of visual representation pre-trained by CoTNet across various downstream tasks, such as object detection, instance segmentation, and semantic segmentation.

5. FINAL DISCUSSION

This preliminary study aimed to address three important issues in the field of deep learning. Firstly, we established the necessity of a blind evaluation set to ensure the reliability and reproducibility of deep learning algorithms. Secondly, we proposed a general deep-learning method for segmenting whole slide images that effectively overcome memory limitations when dealing with large images. Lastly, we introduced a novel vision backbone based on transformer models, which is a prominent area of research in the field of deep learning. These three papers are of significant importance in the context of cancer detection on pathology images, which are typically whole slide images that require accurate segmentation algorithms. Furthermore, deep learning algorithms for cancer detection must be both robust and precise. Therefore, these three papers are fundamental in guiding future research in our group.

Algorithm	Res 2	Res 3	Res 4	Res 5	Params	FLOPs	Top-5 Accuracy
ResNet-50					25.5 M	4.1	93.6
CoTNet-50				✓	23.5 M	4.0	94.1
			✓	✓	22.4 M	3.7	94.3
		✓	✓	✓	22.3 M	3.4	94.4
	✓	✓	✓	✓	22.2 M	3.3	94.5
SE-ResNetD-50					35.7 M	4.4	94.5
SE-CoTNetD-50			✓	✓	23.1 M	4.1	94.7

Table 10. The effects of replacement settings on ResNet-50 and SE-ResNetD-50

6. REFERENCES

Bándi, P., Geessink, O., Manson, Q., Van Dijk, M., Balkenhol, M., Hermsen, M., Ehteshami Bejnordi, B., Lee, B., Paeng, K., Zhong, A., Li, Q., Zanjani, F. G., Zinger, S., Fukuta, K., Komura, D., Ovtcharov, V., Cheng, S., Zeng, S., Thagaard, J., … Litjens, G. (2019). From Detection of Individual Metastases to

Classification of Lymph Node Status at the Patient Level: The CAMELYON17 Challenge. *IEEE Transactions on Medical Imaging*, 38(2), 550-560. doi: 10.1109/TMI.2018.2867350.

Bulten, W., Kartasalo, K., Chen, P.-H. C. H. C., Ström, P., Pinckaers, H., Nagpal, K., Cai, Y., Steiner, D. F., van Boven, H., Vink, R., Hulsbergen-van de Kaa, C., van der Laak, J., Amin, M. B., Evans, A. J., van der Kwast, T., Allan, R., Humphrey, P. A., Grönberg, H., Samaratunga, H., … Consortium, the P. challenge. (2022). Artificial intelligence for diagnosis and Gleason grading of prostate cancer: the PANDA challenge. *Nature Medicine*, 28(1), 154-163. doi: 10.1038/s41591-021-01620-2

Chen, L.-C. C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In V. Ferrari, M. Hebert, C. Sminchisescu, & Y. Weiss (Eds.), *Proceedings of the European Conference on Computer Vision (ECCV)* (vol. 11211, pp. 833-851). Springer International Publishing. doi: 10.1007/978-3-030-01234-2_49.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., & Sutskever, I. (2020). Generative Pretraining From Pixels. *Proceedings of the International Conference on Machine Learning (ICML)*, 1691-1703. url: https://proceedings.mlr.press/v119/chen20s.html.

Chollet, F. (2017). Xception: Deep Learning with Depthwise Separable Convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-8. doi: 10.1109/CVPR.2017.195.

Cruz-Roa, A., Basavanhally, A., González, F., Gilmore, H., Feldman, M., Ganesan, S., Shih, N., Tomaszewski, J., & Madabhushi, A. (2014). Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks. *Medical Imaging 2014: Digital Pathology*, *9041*, 904103. doi: 10.1117/12.2043872.

Dai, Z., Liu, H., Le, Q. V., & Tan, M. (2021). CoAtNet: Marrying Convolution and Attention for All Data Sizes. *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, *5*, 3965-3977. url: https://proceedings.neurips.cc/paper/2021/hash/20568692db622456cc42a2e853ca21f8-Abstract.html.

Deng, J., Dong, W., Socher, R., Li, L., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248-255. doi: 10.1109/CVPR.2009.5206848.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *Proceedings of the International Conference on Learning Representations* (ICLR), 1-21. url: https://openreview.net/pdf?id=YicbFdNTTy.

El-Nouby, A., Touvron, H., Caron, M., Bojanowski, P., Douze, M., Joulin, A., Laptev, I., Neverova, N., Synnaeve, G., Verbeek, J., & Jégou, H. (2021). XCiT: Cross-Covariance Image Transformers. *Advances in Neural Information Processing Systems (NeurIPS)*, *34*, 20014-20027. url: https://proceedings.neurips.cc/paper/2021/hash/a655fbe4b8d7439994aa37ddad80de56-Abstract.html.

Elmore, J. G., Longton, G. M., Carney, P. A., Geller, B. M., Onega, T., Tosteson, A. N. A., Nelson, H. D., Pepe, M. S., Allison, K. H., Schnitt, S. J., O'Malley, F. P., & Weaver, D. L. (2015). Diagnostic Concordance Among Pathologists Interpreting Breast Biopsy Specimens. *Journal of the American Medical Association*, *313*(11), 1122-1132. doi: 10.1001/JAMA.2015.1405.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, *88*(2), 303-338. doi: 10.1007/S11263-009-0275-4/METRICS.

Gao, H. (2023). Understanding the Quadratic Weighted Kappa. Kaggle. [Accessed: 06-Apr-2023]. url: https://www.kaggle.com/code/reighns/understanding-the-quadratic-weighted-kappa.

Guo, Z., Liu, H., Ni, H., Wang, X., Su, M., Guo, W., Wang, K., Jiang, T., & Qian, Y. (2019). A Fast and Refined Cancer Regions Segmentation Framework in Whole-slide Breast Pathological Images. *Scientific Reports*, 9(1), 1-10. doi: 10.1038/s41598-018-37492-9.

Gurcan, M. N., Boucheron, L. E., Can, A., Madabhushi, A., Rajpoot, N. M., & Yener, B. (2009). Histopathological Image Analysis: A Review. *IEEE Reviews in Biomedical Engineering*, 2, 147-171. doi: 10.1109/RBME.2009.2034865.

Hameed, Z., Zahia, S., Garcia-Zapirain, B., Aguirre, J. J., & Vanegas, A. M. (2020). Breast Cancer Histopathology Image Classification Using an Ensemble of Deep Learning Models. *Sensors,* 20 (16)*,* 4373. doi: 10.3390/S20164373.

Hartman, D., Van Der Laak, J., Gurcan, M., & Pantanowitz, L. (2020). Value of Public Challenges for the Development of Pathology Deep Learning Algorithms. *Journal of Pathology Informatics*, 11(1), 7. doi: 10.4103/JPI.JPI_64_19.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. doi: 10.1109/CVPR.2016.90.

Hu, H., Zhang, Z., Xie, Z., & Lin, S. (2019). Local Relation Networks for Image Recognition. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 3464-3473. doi: 10.1109/ICCV.2019.00356.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2016). Densely Connected Convolutional Networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2261-2269. doi: 10.1109/CVPR.2017.243.

Kendall, A., & Gal, Y. (2017). What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? *Advances in Neural Information Processing Systems (NIPS)*, *30*, 1-11. url: https://papers.nips.cc/paper_files/paper/2017/hash/2650d6089a6d640c5e85b2b88265dc2b-Abstract.html.

Khened, M., Kori, A., Rajkumar, H., Krishnamurthi, G., & Srinivasan, B. (2021). A generalized deep learning framework for whole-slide image segmentation and analysis. *Scientific Reports*, 11(1), 1-14. doi: *10.1038/s41598-021-90444-8*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances In Neural Information Processing Systems* (NIPS), 30, 1097-1105. doi: *10.1145/3065386*.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. doi: 10.1109/5.726791.

Li, Y., Yao, T., Pan, Y., & Mei, T. (2022). Contextual transformer networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(2), 1489-1500. doi: 10.1109/TPAMI.2022.3164083.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60-88. doi: 10.1016/J.MEDIA.2017.07.005

Maier-Hein, L., Eisenmann, M., Reinke, A., Onogur, S., Stankovic, M., Scholz, P., Arbel, T., Bogunovic, H., Bradley, A. P., Carass, A., Feldmann, C., Frangi, A. F., Full, P. M., van Ginneken, B., Hanbury, A., Honauer, K., Kozubek, M., Landman, B. A., März, K., … Kopp-Schneider, A. (2018). Why rankings of biomedical image analysis competitions should be interpreted with care. *Nature Communications*, 9(1), 1-13. doi: 10.1038/s41467-018-07619-7.

Nagendran, M., Chen, Y., Lovejoy, C. A., Gordon, A. C., Komorowski, M., Harvey, H., Topol, E. J., Ioannidis, J. P. A., Collins, G. S., & Maruthappu, M. (2020). Artificial intelligence versus clinicians: systematic review of design, reporting standards, and claims of deep learning studies. *British Medical Journal (BMJ)*, *368*. doi: 10.1136/BMJ.M689.

Priego-Torres, B. M., Sanchez-Morillo, D., Fernandez-Granero, M. A., & Garcia-Rojo, M. (2020). Automatic segmentation of whole-slide H&E stained breast histopathology images using a deep convolutional neural network architecture. *Expert Systems with Applications*, *151*, 113387. doi: 10.1016/J.ESWA.2020.113387.

Qin, P., Chen, J., Zeng, J., Chai, R., & Wang, L. (2018). Large-scale tissue histopathology image segmentation based on feature pyramid. *Eurasip Journal on Image and Video Processing*, 1, 1-9. doi: 10.1186/S13640-018-0320-8/FIGURES/4.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (vol. 9351, pp. 234-241). Springer International Publishing. doi: 10.1007/978-3319-24574-4_28/COVER.

Roy, M., Kong, J., Kashyap, S., Pastore, V. P., Wang, F., Wong, K. C. L., & Mukherjee, V. (2021). Convolutional autoencoder based model HistoCAE for segmentation of viable tumor regions in liver whole-slide images. *Scientific Reports*, 11(1), 1-10. doi: 10.1038/s41598-020-80610-9.

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *Proceedings of the International Conference on Learning Representations* (ICLR), 1-14. url: http://arxiv.org/abs/1409.1556.

Srinivas, A., Lin, T.-Y., Parmar, N., Shlens, J., Abbeel, P., & Vaswani, A. (2021). Bottleneck Transformers for Visual Recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16519-16529. doi: 10.1109/CVPR46437.2021.01625.

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2016). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*, 4278-4284. doi: 10.1609/aaai.v31i1.11231.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9. doi: 10.1109/CVPR.2015.7298594.

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In K. Chaudhuri & R. Salakhutdinov (Eds.), *Proceedings of the International Conference on Machine Learning (ICML)* (vol. 97, pp. 6105-6114). PMLR. url: http://proceedings.mlr.press/v97/tan19a.html.

Van Der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., & Yu, T. (2014). scikit-image: image processing in Python. *PeerJ*, *2*(1), e453. doi: 10.7717/peerj.453.

Van Leenders, G. J. L. H., Van Der Kwast, T. H., Grignon, D. J., Evans, A. J., Kristiansen, G., Kweldam, C. F., Litjens, G., McKenney, J. K., Melamed, J., Mottet, N., Paner, G. P., Samaratunga, H., Schoots, I. G., Simko, J. P., Tsuzuki, T., Varma, M., Warren, A. Y., Wheeler, T. M., Williamson, S. R., & Iczkowski, K. A. (2020). *The 2019 International Society of Urological Pathology (ISUP) Consensus Conference on Grading of Prostatic Carcinoma*. *44*(8). doi: 10.1097/PAS.0000000000001497

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I., Brain, G., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. In I. Guyon, U. V Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems (NIPS)* (vol. 30, pp. 6000-6010). Curran Associates, Inc. doi: 10.5555/3295222.3295349.