Deep Learning-based Polyp Segmentation Network with a Dual Encoder-Decoder

by

John Lewis

A thesis submitted to the Faculty of Graduate Studies of

the University of Manitoba

in partial fulfilment of the requirements of the degree of

Master of Science

Department of Civil Engineering

Faculty of Engineering

University of Manitoba

Winnipeg

Abstract

Colorectal cancer (CRC) constitutes one of the most serious forms of cancers across the world. Current screening procedures through imaging with colonoscopy are costly, laborious, and time consuming. Computer-aided diagnosis (CAD) and the integration of deep learning within the field have provided the opportunity for improvements with respect to human error. The recent developments with convolutional neural networks (CNNs) have demonstrated the capacity for real time detection and semantic segmentation of early signs of CRC. These early signs of CRC are referred to as polyps, and thus the identification of these potentially cancerous tumors improves patient outcomes and mortality rates associated with CRC. In an effort to compensate for some of the issues presented for current screening procedures, a novel polyp segmentation network, PSNet [1], is proposed for the semantic segmentation of polyps. PSNet compensates for current issues affecting polyp segmentation networks such as boundary pixel definitions, as well as model generalization and overfitting issues. PSNet provides state-of-the-art (SOTA) performance with respect to two major semantic segmentation performance metrics, mean intersection-over-union (mIoU) and mean Dice (mDice) on 5 major publicly available datasets, with a combined mIoU and mDice score of 0.863 and 0.797 across all sets, respectively. A new configuration of these 5 publicly available datasets is also proposed to improve upon model generalization and help demonstrate some of the issues associated with current standards of polyp datasets and performance evaluation. The mIoU and mDice score for this new configured dataset was 0.941 and 0.897, respectively, and therefore improved even further on the reported SOTA results.

Acknowledgements

I would like to first and foremost express my sincerest gratitude to my advisor in my graduate program, Prof. Young-Jin Cha, who graciously guided me through the past few years and led me through this thesis study. This would not have been possible without his consistent words of encouragement and motivation, along with his patience and understanding. Through Prof. Cha, I would also like to thank Research Manitoba for providing the funding in completing this research study. I also wish to show my appreciation for both of my committee members, Prof. Nan Wu and Prof. Graziano Fiorillo for their assistance and feedback throughout my research study. Finally, I would also like to thank my family and God for being a constant support for me in completing this program.

Table of Contents

List of Tables

List of Figures

Copyright

This thesis is reproduced and extended based on my submitted journal article “Lewis, J. C. & Cha, Y. J. (2022). Dual Encoder-Decoder-based Deep Polyp Segmentation Network for Colonoscopy Images, *IEEE Transactions on Medical Imaging*”. At the time of this submission, the paper is still under review. The corresponding information is referenced accordingly.

Chapter 1. Introduction

In this chapter, rudimentary information pertaining to the field of image processing and deep learning, as well as a general overview of current developments within the field of medical imaging is provided. A synopsis of the literature review conducted for this thesis is also presented, along with the problem definition, objectives of the research study, and the thesis organization.

1.1 Overview

Colorectal cancer (CRC) is one of the leading causes of cancer across the world. This cancer impact parts of the large intestine. Early screening for CRC is essential by the identification of polyps within the digestive tract. Upon localization of these objects, they are biopsied and identified as malignant or benign. Thus, the identification of these polyps is essential. In detecting polyps, CRC can be prevented. Polyps must be identified through colonoscopy procedures, a procedure involving routing a camera through the digestive tract. Given that polyps must be imaged through these invasive procedures, which are both costly and require the presence of medical professionals, image processing has entered into the realm of polyp localization over the past several years.

Image processing is a broad field of computer vision pertaining to the analysis and extraction of features of digital images through the usage of mathematical operations and algorithms [2]. The field of image processing has wide applications in virtually any field of engineering, from the analysis of structures and the improvement of structural health via identification of serious structural abnormalities, to the field of medical image analysis and improvement of patient outcomes via identification of tumours and early signs of cancer.

Medical images can be gathered for effectively any part in the body, and for most locations and organs in the body, the object must be imaged through specific imaging procedures. For

example, colonoscopy procedures are used to identify polyps, adenomas, and tumours in the colon. Images generated from magnetic resonance imaging (MRI) procedures are those that use magnetic fields to generate 3-dimensional (3D) characterization of tissue structures and organs. MRIs can be used for obtaining detailed images of the brain [3]. Computed tomography (CT) scans refer to the methods that generate images from using radiation (such as X-rays) on the body to produce an image of the internal structure and are used to generate detailed images of bones, blood vessels, and muscles [4]. However, each of these aforementioned procedures are subject to human error and potential misses due to variations of whatever abnormality exists in the image sample, such as a tumour or a polyp, be it in the actual identification of such an abnormality or in the full scope of an abnormality within a patient. Moreover, the identification and interpretation of a patient's condition is also time consuming and laborious.

In an effort to solve the problem of human error in medical practice, computer-aided diagnosis (CAD) has been applied to the field of medical image processing. The first description of CAD was defined by G.S. Lodwick [5] in the context of radiology. Since then, the field of CAD has grown. While the success of CAD is contingent on the accuracy of the technology in question, in certain circumstances, the CAD technology can even outperform the diagnosis of experienced medical professionals, for example in the field of radiology and the CAD detection of lung nodules and small tumours as shown in [6].

CAD has also grown with the introduction of machine learning and deep learning-based approaches to CAD. As defined by [7] machine learning is defined as the process of artificially intelligent systems acquiring their knowledge from raw data. Deep learning is the subfield of machine learning in which a sequence of layers or mathematical operations are utilized to obtain higher-level features of data. Convolutional neural networks (CNNs) are a specific application of

deep learning, primarily used within the field of computer vision, that simply use a convolutional operation or layer in at least one or more of the network's layers [7]. CNNs make use of the translational invariance of images, i.e., the property of an image being resistant to system changes with the movement of individual components. Take for example an image of a book, if the book moved 3 pixels to the right in the original image, the picture would still be identifiable as that of a book. CNNs exploit this translational invariance property by convolutional operations and various other mathematical operations which *share parameters* across multiple different image locations [7]. CNNs can also be defined explicitly as a specific form of neural network used for image processing that have a grid-like structure. Another way of viewing CNNs can be that they are structures that are applied to input data and variable parameters within that structure are learned in each of the model's layers, which when tuned and learned, can make predictions on the input with respect to a specific output. The learning and update procedure is done through stochastic gradient descent (SGD), which optimizes and produces a model capable of completing a specific image processing task. The first CNN was developed by [8] with LeNet5. More advanced work in CNNs was provided by [9]. The former of these two examples classified handwritten digits ranging from 0-9, whereas the latter classified 1000 different types of objects with the ImageNet dataset [10].

Within the scope of image processing and deep learning in this thesis, we identify two major image processing tasks related to the field of medical image processing, detection and segmentation. For the purpose of this thesis, detection refers to the identification of the existence of an object. Whereas segmentation we define as the identification of each pixel in an input image belonging to a specific class of object. Segmentation of images is the more computationally expensive task. An example of a network used for detection is shown in [11]. In this network, a transfer learning application was used on an endoscopic dataset to generate decent results in terms

of accuracy, precision, and recall. An example of a more recent network used for polyp segmentation can be found in [12], in which a novel CNN referred to as NeoUNet was used to segment polyps, of which the model was also capable of distinguishing neoplastic polyps.

Datasets such as the Kvasir-SEG dataset [13], the CVC-ClinicDB dataset [14], the CVC-ColonDB dataset [15], the ETIS dataset [16], and the EndoScene dataset [17], have also been developed for the field of polyp segmentation. These datasets have become gold standards in terms of reporting results with respect to the fields of polyp segmentation and detection. Each of these datasets produces a unique set of images of polyps that vary in size, shape, colour, and texture. These are the primary datasets used in the development of the architecture proposed in this thesis and thus will be discussed later in Chapter 4 in Section 4.1.

1.2 Problem definition

Polyps are a key indicator of CRC and thus early identification and removal of these masts are integral to patient mortality. Detection of polyps are currently conducted using imaging procedures conducted by trained medical professionals with little-to-no technological assistance beyond the imaging procedure, which by virtue of being conducted by people, are subject to human error. With the advent of deep learning-based methodologies, automated detection of these features is possible at limited cost, and thus compensation for human errors and reduced miss rates are possible. This research study is an attempt to address the issue of automated detection of polyps on a pixel-based level using a dual encoder-decoder-based end-to-end network. This network is capable of real-time segmentation of polyps and its principal advantage is its accuracy and efficiency.

1.4 Thesis organization

This thesis is comprised of 6 chapters. Chapter 1 identifies the scope of the problem and research direction. In Chapter 2, we provide a detailed literature review relating to the research topic of polyp segmentation. In this chapter we go over key details of CRC, along with imaging procedures, characteristics of polyps, and important statistics pertaining to CRC. These details are presented in the hopes of identifying the primary motivation behind this research study and the impacts of the proposed research. We also go over key research developments in the field of deep learning that attempt to provide innovative solutions for these problems. In Chapter 3, we describe the methodology corresponding to the proposed architecture for polyp segmentation, PSNet. In this chapter we will discuss the key components of PSNet, a detailed review of the proposed and generic modules implemented in its construction, along with a discussion of the loss function chosen for the model. In Chapter 4 we discuss the datasets used and the implementation details used for model training, such as the learning rate scheduler and the optimizer chosen. We also briefly go over key machine learning concepts such as SGD and their mathematical formulation. In Chapter 5 we discuss case studies and comparative studies evaluating the performance of our proposed network. Finally, in Chapter 6, we provide a conclusion and summary of this thesis, discuss current limitations of the model, and give a brief proposal for future work.

1.3 Objectives

One primary objective of this research study is to improve upon the automated detection of polyps on a pixel-based level, i.e. the semantic segmentation of polyps. This research study seeks to improve upon model accuracy through a variety of common performance metrics, such as mean intersection-over-union (mIoU) and mean Dice (mDice). Another objective is also to produce a

model architecture capable of *real-time* segmentation of polyps. However, the ultimate goal of this research study is the improvement of patient outcomes by means of early detection of CRC. The software produced in this research study could be used for hardware applications in real-time and thus compensation for human error endemic to endoscopic imaging procedures. Alternatively this software could be used for post hoc reviews of record colonoscopy data to reiterate the finding of a polyp or not. This research study employs the following measures to accomplish these objectives:

- Primary collection of data via colonoscopy procedures. Several publicly available datasets are utilized to complete this requirement, each with a diverse range of polyps ranging in different shapes, textures, and colours.
- The datasets are amalgamated and split into training, validation, and test sets to evaluate the model performance.
- A novel dual model structure was developed to identify, on a pixel-based level, healthy tissue from polyps.
- This architecture was created to improve upon model generalization issues associated with previous models and boundary pixel limitations of previous polyp segmentation networks. This model consists of a unique arrangement of novel modules that can be deployed for a variety of different general usages within the field of image processing (hopefully even outside the domain of medical image segmentation as well).
- The model performance is evaluated against other pre-existing polyp segmentation networks and industry-standard models to compare performance and establish state-of-the-art (SOTA) results.

Chapter 2. Literature Review

In this chapter we discuss a thorough review of CRC and its impacts on the general population, going through incidence rates, prognosis, and mortality. This chapter aims to provide a primary motivation of this research study. We also provide a thorough discussion of architectures foundational to CNN theory and early works that were pioneered in the field. In addition to these topics, the two major fields of image processing pertaining to polyps are discussed, that of polyp detection and polyp segmentation.

2.1 Colorectal cancer (CRC)

As stated previously, CRC is a form of cancer that primarily impacts the colon and the rectum. A severe malignancy, CRC constitutes the third most common form of cancer for both women and men [18], [19]. CRC also constitutes one of the most severe forms of cancer and cancer related mortality with over 1 million new cases of CRC diagnosed across the world in 2002 [19]. Prognosis varies significantly around the world and depends on access to medical services, early screening, as well as factors such as age, sex, smoking, etc. Primary risk factors associated with CRC are medical and family history, along with presence of colorectal polyps and inflammatory bowel diseases [20].

Principle diagnosis of a polyp is found by obtaining a colon sample from either a sigmoidoscopy or a colonoscopy. These methods constitute the main screening procedures for CRC. Screening procedures for CRC are recommended to begin at age 50. Polyps are small masts within the digestive tract that can be an indicator of CRC, however they can also be benign. The most common form of polyp is adenomatous polyp and approximately 10% of these polyps will slowly develop into cancer. CRC gradually metastasizes into cancer over 10-15 years and will typically begin as a noncancerous polyp [20]. Polyps vary in size, shape, colour, and texture and

can be divided into a number of classes of polyps, such as simpler hyperplastic, adenomas, and cancerous [21]. Approximately 90% of colorectal polyps are smaller than 1 cm, and approximately 80% of polyps are smaller than 5 mm, and are often noncancerous. Given that the rate at which CRC develops is slow, identification of polyps frequently results in the mere dating of a follow-up assessment [22]. However, the identification and removal of these polyps has been associated with a higher prognosis and reduction in CRC as shown by [23], which demonstrates that a 53% mortality reduction is possible through identification of polyps through colonoscopy procedures.

One measure of CRC progression and prognosis is the 5-year survival rate, indicating the likelihood of a patient to live past five years after diagnosis of CRC. 5-year survival rates for this form of cancer vary, however an average estimate across all types of CRC can be estimated between 48.6% and 59.4% [24]. It should be noted that mortality rates associated with CRC have been decreasing over the past several years. For example, in the United States, CRC mortality rates decreased by 34% for individuals aged older than 50 years between 2000 and 2014 [25]. This can be ascribed to changes in societal habits such as a lower incidence rate of smoking, non-steroidal anti-inflammatory drugs (NSAID) intake, and red meat consumption. However, [25] also notes that the incidence rates of CRC increased for individuals below the age of 50 by 13%, over this same period and also reports that the overall incidence rate of CRC across all persons in the United States was 48.6 per 100,000 persons. In addition to an increase in CRC amongst persons under the age of 50, miss rates associated with polyps have been estimated at 15-30% for small polyps at diameters less than 5 mm [26]. Thus, the necessity of identifying CRC and improving clinical and patient outcomes is paramount.

2.2 Early work in CNNs and fundamentals

CNNs made their first appearance in [8] with the primitive convolutional neural network, LeNet5. This primitive CNN structure was initially used to classify handwritten digits. Hence, the images and the ground truths consisted of a total of 10 different classes, with each image and ground truth being a digit ranged between 0-9. The input image size was constrained to $32 \times 32 \times 1$ with respect to the height and width and was single channeled (i.e., black and white). The network consisted of 5 layers, each with learnable parameters. The model consisted of 3 convolutional layers, each followed by average pooling. After the sets of convolutional layers and the average pooling layers, two fully connected layers were implemented, followed by a SoftMax layer which ascribed the input image a class or predicted digit.

However, little work was put forward into CNNs until the development of AlexNet by [9]. Similar to [8], the task within image processing that AlexNet was first used for was detection. Originally published at the LSVRC-2010 contest, the structure provided its form of accuracy in terms of an error rate, which the final reported value was 15.3%. Indicating that 15.3% of samples were classified incorrectly of the 1.2M images within the ImageNet dataset. This value was significant as it improved upon the existing record of 26.2%. The network took in image sizes of $224 \times 224 \times 3$. With the channel dimension of 3 indicating that the image took in RGB images. The loss function used for training was categorical cross entropy (CCE). The base structure of AlexNet contained 5 convolutional layers, 3 maxpooling layers, 3 dense (also known as linear or fully connected layers), along with a SoftMax layer at the end. Sigmoid activations were used between convolutional layers as the nonlinearity. The model was trained through SGD and the filters or kernels within each convolutional and fully connected layer were updated and learned to train the model. The model was trained for 90 epochs and the training time took 5 to 6 days.

The impact of AlexNet on the field of machine learning cannot be overstated, evidenced by the fact of the wide variety of fields that this architecture has been found in since then. For example, with crack detection as demonstrated by [27], as well as with breast cancer diagnosis with histopathology images as seen in [28]. Beyond AlexNet in 2012, further advances were made in the field of object detection with CNNs. Other landmark architectures frequently found in some form within contemporary CNNs used for classification were GoogLeNet and VGGNet, both developed in 2014, by [29] and [30], respectively, in which both architectures improved upon the classification accuracy of AlexNet.

Following the success of classification and detection tasks in image processing with CNNs, inroads were made within the field of semantic segmentation. Primary inroads in semantic segmentation were made by [31]. This network built off the VGG16 network for its best results by replacing all the dense layers with convolutions, thus obtaining more spatial data. By introducing a pyramid structure and deconvolutional layers to produce an output segmentation map, pixel-level detection was possible. This network achieved an increase of more than 20% with respect to the existing benchmarks and records held in the field of semantic segmentation, thus further reiterating the ability for CNN architectures to dominate image processing tasks. The network also explored semantic segmentation networks with AlexNet and GoogLeNet which performed more poorly relative to the fully convolutional network used with VGG16 as the backbone.

Major inroads were made within the field of medical image segmentation with UNet. UNet is a convolutional-based deep learning method that was originally developed for segmenting neuronal structures and cell tracking produced by images generated from electron microscopes [32]. Winning the best paper in 2015 for the ISBI cell tracking challenge, UNet has become a de facto standard for comparative studies and baseline architectures within the field of medical image

segmentation and general image processing. The architecture is a pyramid-based structure, hence the aptly named “U”. The architecture is widely described as an encoder-decoder-based structure. An encoder can be described as a network that processes the input to a lower dimensional form, obtaining spatial information on the feature map through model training. Whereas the decoder processes that lower dimensional form back up to the same input dimension size (with a variable number of channels, corresponding to the number of classes), producing a prediction on the input feature map [7]. The UNet architecture consists of 4 successive 3×3 double convolutions, each followed by batch normalization (BN) and rectified linear unit (ReLU) activations, then followed by downscaling with maxpooling operations, and thus extracting higher-level feature information. The model then moves into the second part of the pyramid which contains 4 successive up-convolutions (upsampling followed by double convolution, with the latter identical to the double convolutions found in the encoder). The final layer provides a pointwise (PW) convolution and converts the image to the output segmentation map dimensions, with the number of channels indicating the number of classes the architecture attempts to classify and learn. The original architecture took in images of input size $572 \times 572 \times 3$.

Currently, many CNN-based segmentation networks feature encoder-decoder-based architectures as seen in [32], [33], and [34]. The architecture proposed in [33] contained an encoder-decoder based architecture referred to as DeepLabv3+, which builds off the existing architecture DeepLabv3 [35]. The new addition to the model built off the existing atrous spatial pyramid pooling module by supplementing it with depthwise separable convolutions, while also implementing a new decoder module capable of recovering more object boundaries. The model is evaluated on the Cityscapes dataset [36], a benchmark dataset for general image processing tasks. The model achieved SOTA results with respect to both of these datasets with a test performance

of 89% on the Cityscapes dataset. The model's most notable contributions however were their employing of spatial pyramid pooling and atrous convolutions, which are able to obtain contextual information at various resolutions.

2.3 Polyp detection

With the rapid rise of semantic segmentation via the advent of modern computers and the consequent lower computational cost for heavier image processing tasks associated with CNNs, the work of the mere detection of polyps has become virtually obsolete. Clearly indicated by the fact that the identification of a polyp is implied by the segmentation of an image and identification of regions of interest (ROIs) containing polyps. That being said, early work into CNNs was primarily detection focused. An example being that by [37] who produced a small AlexNet-like structure. The model proposed contained 2 convolutional layers and 2 pooling layers and classified an individually procured dataset of 1,200 images as either adenomatous or non-adenomatous (i.e., whether containing a polyp or not). The image input size to the model was $256 \times 256 \times 3$ and the accuracy reported by the model was 75.1%.

Another example of an AlexNet-like architecture used for polyp detection was performed by [38]. The model was composed of 3 convolutional layers, 3 maxpooling layers, and a fully connected layer. Similar to other detection models, the model classified the input features as either containing a polyp or not. The dataset used was obtained from the ISBI 2015 Grand Challenge on Automatic Polyp Detection. The accuracy reported for this model was 90%, meaning 90% of the test dataset for the challenge was evaluated correctly in its identification of a polyp or not.

Another example of early work in polyp detection was performed by [11], in which a transfer learning application was developed followed by an ensemble network. Transfer learning is the process within deep learning in which a model is trained on a separate dataset (in this case,

a large non-medical dataset) and then using those subsequent trained weights within another (new) model that is then used for another task (in this case, polyp detection). This process assists in convergence and avoids instability that may occur during training. The process is now a common practice within the field of deep learning and is particularly useful when training on a larger dataset is required. For the non-medical images used for pretraining, the model was trained on ImageNet [10] and Places205 [39]. The two were chosen for the high degree of overlap between classes within each dataset, improving upon variation and model exposure to different conditions. For the images used for polyp segmentation, the model used an original unbalanced dataset containing endoscopic images, meaning not all images within the dataset contained an identifiable polyp, and that the division of images containing polyp and endoscopic image without a polyp was not equal. The number of endoscopic images without polyps was 1104. Both hyperplastic polyps and adenomatous polyps were present in the dataset, with 363 images of hyperplastic polyps and 563 images of adenomatous polyps, respectively.

The model consisted of 4 convolutional layers followed by a support vector machine (SVM), which divided the output into a terminus describing non-polyp and another branch containing a structure to further classify the two kinds of polyps. The second branch continued on, containing 3 more convolutional layers followed by another SVM which further divided the classification between hyperplastic polyp and adenomatous polyp. The final reported accuracy for the model was 86.9%, the final reported precision was 87.3%, and the final reported recall was 87.6%. This method can also be qualified as an ensemble method due to the fact that it incorporates both an SVM and a CNN. These methods can be very useful by exploiting the properties or advantages of both, while mitigating their shortcomings. Ensemble methods will be discussed

further in this chapter in Section 2.4 in the context of ensemble methods used for polyp segmentation.

Since then, there have been other polyp detection models such as the YOLOv2-based model proposed by [40]. The architecture consists of 19 layers of either maxpooling or convolutional layers, with each convolutional layer followed by BN and ReLU. The final layer is a convolution to the number of output classes, in which it was originally trained on 1000. Transfer learning was then implemented and the pretrained weights were applied to the new model in which the total number of output classes was 2 (polyp or non-polyp). The model evaluated its results on 3 datasets. One of the datasets is currently publicly available, the CVC-ClinicDB dataset [14], and the sensitivity reported was 90.2%. The model was also evaluated on 2 internally collected datasets and the sensitivities for these datasets were 96.7 and 87.7%, respectively.

In [41] an automated polyp detection network was developed using a 3-dimensional (3D) fully convolutional network (FCN). The model integrated both spatial and temporal features into the learning of their 3D-FCN through the usage of colonoscopy videos, while also providing an online and offline version of the model. The online version improved upon the tendency towards false positives of the offline version, thus further improving the accuracy of the model. This model attained SOTA accuracy with respect to the F1 and F2-score.

2.4 Polyp segmentation

Outside of the realm of deep learning, [42] performed a contour analysis to obtain results with respect to the detection and segmentation of a polyp, obtaining a 90.5% accuracy with respect to

polyp detection, as well as 71.57% with respect to their Dice coefficient which evaluated their segmentation score.

Ensemble methods can be defined as the usage of multiple different machine learning architectures or exclusively CNN-based architectures that are used in tandem with one another to achieve a specific image processing task, such as segmentation. Intuitively, these methods have the capacity to increase the accuracy and model performance beyond the individual accuracies of each of their individual components. Ensemble methods have been used frequently over the past several years, such as in the works of [43], [44], and [45].

In [43], the existing architecture, Mask R-CNN, was applied to the explicit task of polyp segmentation. They employed an ensemble method, a dual-mask R-CNN, which was used to perform the same task and improve the model performance. The structure operates the two Mask R-CNNs synchronously and then performs a bitwise combination on the outputs of each of the two models to generate the final output segmentation map. The authors also employ transfer learning principles by using pretrained weights from the COCO dataset onto the model [46]. The models then evaluate their datasets on three open-source datasets previously mentioned, the CVC-ClinicDB [14], CVC-ColonDB [15], and ETIS [16] datasets. The authors report their results on the latter two datasets. These model's significantly outperformed SOTA models, with an mIoU score of 66.1% and 69.5% for the ETIS and CVC-ColonDB datasets, respectively.

In [44], the authors construct a tri-model approach, in which UNet-VGG [32] [30], SegNet-VGG [30] [47], and PSPNet [48] (all existing architectures used for general image segmentation tasks) were used to produce an ensemble architecture capable of polyp segmentation. The authors used a weight voting method to obtain the final output segmentation map. Similar to other contemporary, competitive models, the authors evaluate their results on 3 publicly available

datasets, EndoScene, CVC-ClinicDB, and the ETIS datasets. From these datasets, the authors increased their training, testing, and validation size 10-fold by applying augmentation to the images and generating new input images for the corresponding sets. In [45], the authors use a combinatory approach forming a model that uses both a DeiT transformer [49], as well as a ResNet structure [50]. This model also employed a fusion module to integrate feature maps at different resolution levels, as well as utilized deep supervision with these output feature maps from the fusion modules. The model achieved SOTA however had a higher model complexity.

The authors in [51] conducted extensive review and comparative studies by evaluating many different existing CNN-based architectures to examine segmentation and classification performances of endoscopic objects. The authors detail 23 different algorithms consistently used in segmentation and detection tasks and evaluate their respective performance for the Endoscopy Artefact Detection (EAD) Challenge. This dataset is succinctly different from previously discussed polyp segmentation sets, as it contains information pertaining to a variety of different artefacts present in endoscopic images and settings, such as specularity, saturation, artefact, blur, contrast, bubble, and instrument. These artefacts can hinder quantitative analysis by obstructing ROIs. Thus, the number of classes in this task is no longer 1 (which indicates either healthy tissue or polyp), the number of classes is equal to 5.

The authors note that no specific method that they evaluated outperforms across all tasks (i.e., detection, segmentation, and out-of-sample generalization). Thus, what we can infer from the findings of this paper is the notion that segmentation networks are designed for very specific tasks, and that when implementing CNNs for semantic segmentation, that maximum performance is achieved by confining the scope of the model to a specific task or a specific set of classes, such as polyps. Also, that SOTA segmentation networks perform best across a large number of those

specific, confined image processing tasks and generally if one performs well on one specific dataset, it will perform well on another. Moreover, that CNNs are spatially invariant and that their ability to generalize is specific to the quality and quantity of the dataset and not necessarily specific properties of the class of object, i.e., given an adequate dataset with decent lighting of dogs, a model could produce the exact same levels of accuracy given a similar quality and quantity dataset of cats.

Currently, the vast majority of polyp segmentation networks feature an encoder-decoder-based structure, such as in [34] and [52]. As touched on previously in the discussion of [32], encoder-decoder frameworks work by progressively downscaling an image through convolutions and maxpooling operations, while integrating other mathematical operations such as activations or nonlinearities. This portion constitutes the encoder and learns by progressively extracting higher-level information from the features in the convolutional operations and gathering short-range semantic information. The decoder portion consists of a progressive increase in resolution size through upsamplings or transposed convolutions, or any other operation that increases the feature map size, while potentially also integrating previous feature representations with skip connections. A potential issue to be discussed is the defining aspect of CNNs in which they are limited to local-level features through the convolution operation. Increasing the size of the convolutional kernel to potentially gather more global contextual relationships comes at an exponential increase in the model complexity. This is the primary reason behind instituting skip connections, as it is a more economical integration of global feature information back into the architecture that would have otherwise been lost through progressive dimensionality reductions through convolutions and maxpooling operations.

ColonSegNet is presented in [34], which is an encoder-decoder architecture and features a ResNet block [50] with squeeze-and-excitation [53], with two encoder blocks and two decoder blocks. ColonSegNet, by utilization of the squeeze-and-excitation modules, has considerably less learnable parameters and thus improves upon computational complexity and runtime. Similar to other models, the network also features skip connections, which through concatenation along channel dimensions, integrate features from earlier in the network that would otherwise be lost through progressive convolutional and maxpooling operations that decrease the feature size. Thus, the network also has a capacity to learn global dependencies. The model reported a competitive frames-per-second (FPS) statistic at 180.00 FPS, as well as 72.4 IoU score reported on the Kvasir-SEG dataset [13].

In [52], PolypSegNet is proposed. This model implements a proposed depth dilated inception (DDI) module, a deep fusion skip module (DFSM), as well as a deep reconstruction module (DRM). In the DDI module, dilated convolutions were used which can increase the accuracy by the increased receptive field that exists within the dilated convolutional kernel. The model had a competitive number of parameters, 5.5M, which is significantly lower than standard architectures such as UNet, which in a standard configuration has 7.8M. However, this model assumably trained its results *individually* on each of the publicly available polyp datasets previously discussed, and not any standardized dataset or structuring of those datasets therein. Thus, the reported model performance statistics are hard to compare with more contemporary SOTA models.

Another example of an encoder-decoder-based structure, as well as that of an ensemble method, used for polyp segmentation is that produced by [54]. In this FCN architecture, the input images of the polyps were originally inserted into a binary classifier which was designed to make

an initial prediction as to whether there was a polyp present in the image or not. The binary classifier extracted global characteristics, such as the ColorLayout, EdgeHistogram, and the AutoColorCorrelogram, to make the assessment if a polyp was present. If a polyp was present in the image, the input feature map proceeded to the FCN component and the image was segmented. This FCN component was similar to UNet as established by [32]. The network attained SOTA sensitivity and specificity performance with respect to the Kvasir-SEG and the CVC-ClinicDB dataset.

Beyond the issues of extracting global dependencies, other problems CNNs are subjected to are overfitting and the ability to obtain information on boundary pixels. Generally speaking, boundary pixels are often poorly defined and as a consequence are the more difficult regions for a CNN to learn. In recent years, there have been attempts to compensate for these problems, such as with PraNet, developed by [55]. PraNet provided a number of proposed modules aimed at explicitly capturing boundary pixel information. Specifically, this module implemented a reverse attention (RA) module which worked by first getting rid of background information and consequently mined the polyp boundary region. This operation was implemented with the use of a reverse-attention weight [56]. The RA then works by establishing boundary cues which help define the relationship between healthy tissue and polyp, along with the boundaries between them. PraNet also implemented a parallel partial decoder (PPD) which integrates higher-level features, which require less computational resources and contribute more to model performance. This model achieved SOTA performance with respect to mIoU and mDice.

Another example of an attempt to address boundary pixel limitations is the Learnable Oriented-Derivative Network (LOD-Net) put forward by [57]. In this network, the model obtains information on boundary pixels by using oriented derivatives. As the paper suggests, oriented

derivatives are larger for boundary pixels and thus can be calculated for regions to hone in on the locations of boundaries. The paper also used a backbone to obtain higher-level semantic features. The model evaluated itself with the 5 major publicly available polyp datasets and achieved SOTA performance with respect to Kvasir-SEG, ETIS, CVC-ColonDB, and the EndoScene datasets.

Using principles similar to PraNet [55], AMNet was proposed by [58], a recent SOTA network for polyp segmentation. This network also employs a PPD, which as in PraNet, also aggregates higher-level features. The model also proposes two key modules, the channel-wise feature pyramid (CFP) module and the polarized self-attention (PSA) module. The CFP module aggregates higher-level feature information by combining the outputs from progressively increased dilated convolutions, which allow for a broader context to be retrieved. The PSA module is similar to the squeeze-and-excitation modules as proposed by [53]. The model achieved SOTA accuracy with respect to 4 of the major publicly available polyp datasets, Kvasir-SEG, CVC-ClinicDB, CVC-ColonDB, and ETIS.

Of particular importance to the field of CNNs has been the introduction of deep supervision into polyp segmentation networks, as well as generalized medical image segmentation networks. It is an approach used within deep learning primarily used to improve performance in terms of convergence time. Deep supervision, originally proposed by [59], has been employed in a number of polyp segmentation networks, such as in [45], [55], and [60], as well as into the field of segmentation pertaining to the task of cancer detection in breast ultrasounds as seen in [61]. Deep supervision can be simply described as the incorporation of coarser representations from the network producing output segmentation maps at each representation, and then consequently including those output segmentation maps in the loss function or the final output segmentation map. This approach ameliorates the convergence of the network and reduces training time.

Chapter 3. Methodology

In this chapter the methodology corresponding to the structure of the proposed architecture used for polyp segmentation is presented, along with a discussion of the choice of loss function, as well as the training and implementation details. The training and implementation details cover the optimization method, the learning rate scheduler, and other concepts and artefacts used in training the network.

3.1 Overview of the proposed network

The proposed polyp segmentation network, PSNet, has been newly designed for the specific task of polyp segmentation. The model is composed of a dual encoder-decoder structure, with a variety of different novel and generic components. An overall schematic of the network architecture is provided in Figure 1. Designed with maximum performance at an input RGB image size of 512×512, the network has been trained thoroughly and tested against a number of publicly available polyp datasets, retrieved from stills in colonoscopy videos. The model is generalized and has the capacity to identify and segment polyps ranging in a wide variety of shapes, sizes, colors, and textures. Thus, PSNet is capable of differentiating polyps from normal tissue and has clear clinical applications.

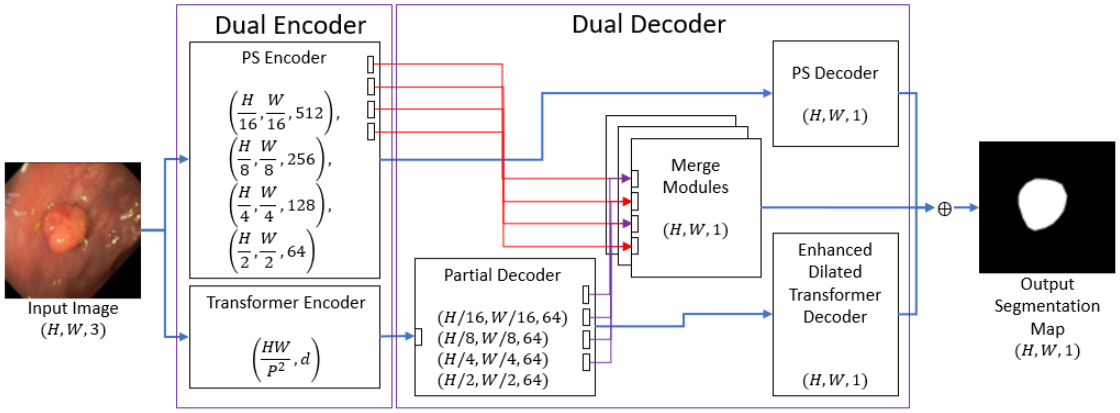

Figure 1. Schematic diagram detailing overall structure of PSNet

PSNet's dual encoder structure is comprised of a convolutionally based encoder, referred to as the PS encoder, as well as a generic transformer-based encoder which we simply just refer to as the transformer encoder. The two encoders receive the same input image and process it to a lower dimensional form with feature extraction modules. Both encoders run simultaneously, and both contain attention mechanisms such as skip connections with the PS encoder and the multi-scale self-attention (MSA) module in the transformer encoder. Following the feature extraction that occurs in the dual encoder, the feature maps are then put through the dual decoder component. The dual decoder component consists of the PS decoder, merge modules, and the enhanced dilated transformer decoder.

The primary function of the decoder is to generate output segmentation maps and classify, on a pixel-based level, the location of polyps within an input image. A generic partial decoder is used to convert the output of the transformer encoder to a 3D tensor, before being passed to the enhanced dilated transformer decoder to produce a candidate output segmentation map. The PS decoder receives input from the dual encoder as well as coarser input feature maps from different resolutions in the PS encoder. Deep supervision is employed through the implementation of the merge modules, which individually produce candidate output segmentation maps. Inputs to the merge modules are indicated by the red and purple lines in Figure 1. A total of 6 candidate output segmentation maps are generated from the PS decoder, enhanced dilated transformer decoder, as well as 4 from the merge modules. These candidate output segmentation maps are then averaged, as indicated by the $\oplus$ in Figure 1, and a final output segmentation map is produced. The aforementioned components will be discussed in greater detail in their corresponding sections below.

A dual model structure comprised of both transformer-based and CNN-based architectures is advantageous due to the nature of each of the two approaches in deep learning architectures. CNNs are limited inherently to local-level features, whereas transformers distinguish themselves by their ability to capture global dependencies and long-range semantic information. CNNs are limited to local-level features due to the limitations of the convolution operation, which while advantageous for images which are translationally invariant, provides an overall limitation on the maximum accuracy that can be achieved in architectures that are fully convolutional. Therefore, the impetus behind combining the two approaches is apparent: by establishing a mutually and synchronously working architecture that harnesses the power of both transformers and CNNs, it can be expected that the model accuracy and performance will be increased, as discussed previously with respect to ensemble networks. PSNet's dual structure thus works by having the PS encoder work on and learn local-level phenomena, and the transformer encoder works on and learns global features and context. The one trade-off associated with this increased accuracy is a reduced runtime due to the increased number of parameters. However, comparing PSNet to other ensemble or dual model methods that employ a similar approach have a lower number of parameters relative to PSNet.

3.2 Dual encoder

The dual encoder featured in PSNet is a newly crafted, simultaneous convolutionally-based and transformer-based module. The module is composed of both a CNN-based encoder and a generic transformer encoder. The former is referred to as the PS encoder, the latter is referred to as the transformer encoder. The generic transformer encoder is the VisionTransformer (ViT) provided in [62], whereas the CNN-based encoder is a novel module produced for the explicit purpose of polyp

segmentation. Given that the PS encoder is CNN-based it is translationally invariant and has a locally restricted receptive field throughout its structure. Whereas, the generic transformer lacks these inductive biases and is therefore translationally variant and the receptive field is global. Thus, by design, each sub-encoder within the dual encoder works on particular regions and scales. PS encoder focuses on local phenomena and the transformer encoder focuses on global context. The PS encoder's receptive field is expanded through the usage of dilated convolutions and newly developed modules that enhance the local feature extraction needed for this component of the dual encoder.

Both encoders receive the same input image. However, the PS encoder produces an output at feature scales $1/16^{\text{th}}$ of the input image's height and width dimension and proceeds directly to the PS decoder. Whereas the transformer encoder produces a 2D output that needs to be reshaped to a 3D tensor before proceeding to the enhanced dilatated transformer decoder to produce an output segmentation map. This process is handled by the partial decoder. In terms of relative contributions to model complexity, it should be noted that the transformer encoder contributed the most to the total number of parameters of the model. This makes sense instinctively on the basis that modelling global dependencies is more expensive than modelling local dependencies.

3.2.1 PS encoder

The PS encoder is a newly developed CNN-based encoder that is designed for local feature extraction. The PS encoder performs this task extremely efficiently. The PS encoder takes in an input image of dimension, $x \in R^{H \times W \times 3}$, and applies 4 sequential maxpooling operations to reduce the dimensions of the image and obtain lower-level features from the subsequent convolutional operations. The PS encoder produces an output, $x \in R^{\frac{H}{16} \times \frac{W}{16} \times 512}$, before subsequently being fed as input to the PS decoder as well as the merge modules. The PS encoder also outputs coarse

representations through skip connections to the PS decoder, as well as to the merge modules to be fused with the output from the transformer encoder and partial decoder. At half the image input size, an output feature map, $x^{1/2} \in R^{\frac{H}{2}\times\frac{W}{2}\times 64}$, is used as input for both a merge module, and a skip connection. This same sequence of operations is true for the output feature maps, $x^{1/4} \in R^{\frac{H}{4}\times\frac{W}{4}\times 128}$ and $x^{1/8} \in R^{\frac{H}{8}\times\frac{W}{8}\times 256}$ produced by the PS encoder. A diagram describing the architecture of the PS encoder is shown in Figure 2.

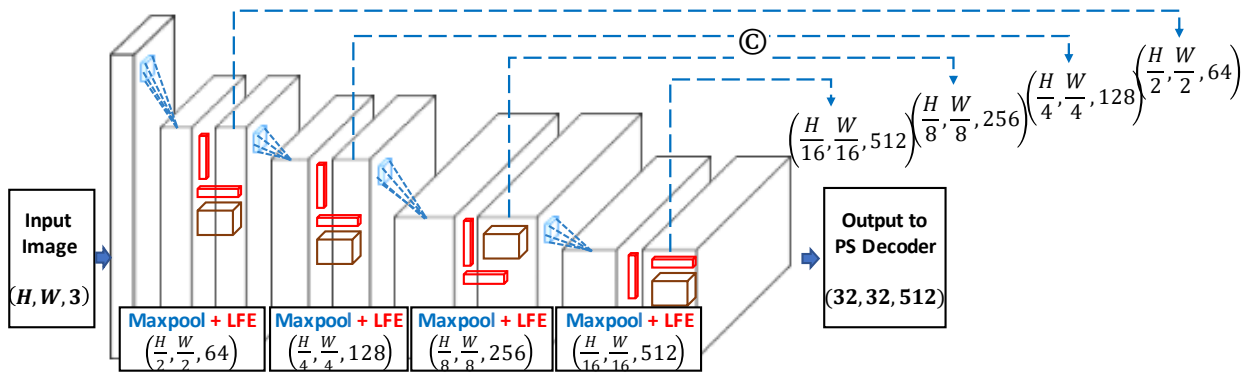

Figure 2. PS encoder detail [1]

The kernel size chosen for the maxpooling operation was 2 for all maxpooling layers. The maxpooling operation is an operation without weights or learnable parameters, thus it reduces the model complexity, and improves the model runtime. Maxpooling can also assist in reducing overfitting and it can intuitively be thought of as returning the most important images within the kernel that slides through the input feature map within the layer. Following each maxpooling operation is the local feature extraction (LFE) module, which has also been developed for this explicit polyp segmentation framework. Part of the design focus of the PS encoder was to learn the features and representations of small polyps, which was achieved by small kernel sizes in both the maxpooling and convolutional layers.

While discussion of every mathematical operation and layer within the structure is out of the scope of this thesis, a brief discussion will be provided on the separable convolutions present in the PS encoder (and also the merge modules). Every convolutional operation in the PS encoder (as well as the merge module) is separable. Separable convolutions are performed significantly faster in comparison to typical convolutions. These convolutions expedite the operation by having lower computational complexity when compared to standard convolutions as shown in [63]. This is done by sequentially applying a depthwise (DW) and pointwise (PW) filter, as shown in [64]. These operations are summarized in Equations (1) and (2).

$$\mathrm{PW}_{\mathrm{Conv}}(\mathrm{w},\mathrm{x})(\mathrm{i},\mathrm{j}) = \sum_{\mathrm{c}}^{\mathrm{C}} \mathrm{x}_{\mathrm{i,j,c}}\mathrm{w}_{\mathrm{c}} \tag{1}$$

$$\mathrm{DW}_{\mathrm{Conv}}(\mathrm{w},\mathrm{x})(\mathrm{i},\mathrm{j}) = \sum_{\mathrm{u,v}}^{\mathrm{k,k}} \mathrm{x}_{\mathrm{c,i+u,j+v}}\mathrm{w}_{\mathrm{u,v}} \tag{2}$$

where w and x are the weights and input, respectively. i and j are the coordinates corresponding to the height and width dimension, and c is the coordinate corresponding to the channel dimension. The DW component is a spatial convolution applied over each channel, whereas the PW component then projects the output of the channels in the DW component into a different channel space, which as described by [63] model complexity is reduced.

3.2.1.1 LFE Module

The proposed LFE module is newly developed for this research study and incorporates a multiscale approach to local feature extraction. The LFE module is shown in Figure 3. The LFE block is the foundational unit of the PS encoder and is comprised of two components within it, a dual separable convolution (DWSC), as well as a dual complex convolutional module (CCM). With the latter of

the two components consisting of largely dilated convolutions, which serve to enhance the context of the local features extracted in the LFE module. The number of output channels, C', is kept constant throughout the LFE module and is a tunable hyperparameter, in that the selection of this value can be modified. Consequently, increasing or decreasing its value needs to be optimized through successive iterations of model training with varying values of the parameter. We found optimum performance of this value at C' to be 64, 128, 256, and 512 for each LFE module implemented in the PS encoder, as reflected in Figure 2.

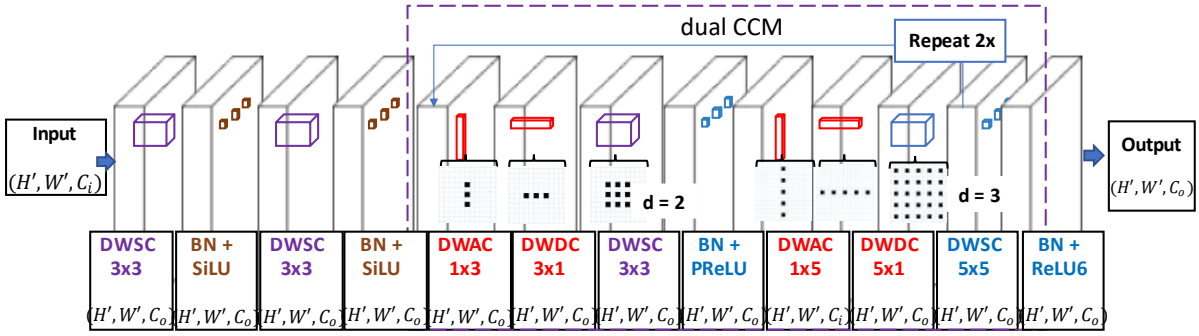

Figure 3. LFE module detail [1]

The LFE module receives an input feature map, $x \in R^{H' \times W' \times C'}$, and initially processes the input feature map through a sequence of two DWSC modules. Each convolution is followed by BN and a sigmoid linear unit (SiLU) activation in the DWSC modules. The SiLU function is similarly shaped to the ReLU function, the latter of which is frequently observed as a choice of nonlinearity in contemporary CNNs. ReLU is primarily chosen due to its simple derivatives in backpropagation in gradient descent, resulting in lower computational costs. Given that the SiLU function is similarly shaped to the ReLU function, it exhibits similar properties with respect to its output in forward propagation, as well as its derivatives in backpropagation. However, the SiLU

function has smoother curvature and shape when compared to the ReLU function, while also maintaining stability in training. In our own parametric studies investigating activation functions for the LFE module, it was determined that that the SiLU function performed the best in terms of model accuracy, while maintaining approximately equal training time relative to other activations investigated, such as ReLU.

The kernel size, stride, and padding for each of these DWSC modules was 3×3, 1, and 1 respectively. This configuration of kernel size, padding, and stride results in the dimensions of the output from these two modules being the same as the dimensions of the input, i.e., the dimensions remain constant throughout the LFE module. We leave the dimensionality reduction within the PS encoder instead to the maxpooling module. This aids in PSNet not overfitting by reducing the number of learnable parameters (if the dimensionality reduction occurred through larger kernel sizes and strides, the number of parameters would increase). Moreover, there is a possibility of losing important features due to drastic reductions in spatial size of the output feature map. Thus, in maintaining principles of capturing local-level features, a smaller kernel size of 3×3 throughout the PS encoder and single unit stride and padding was selected.

Within the LFE module, following the DWSC modules is the dual CCM submodule. The CCM submodule consists of a depth-wise asymmetric convolution (DWAC) and a depth-wise dilated convolution (DWDC). The module is based off the ICM submodule proposed by Ali & Cha [65]. The primary difference between the ICM submodule and the CCM submodule is that in this context, the CCM's sequence of operations is repeated sequentially as opposed to synchronously and concatenated in the ICM. The primary purpose of the CCM submodule is to attend contextual features and complex correlations present in the input feature map.

The dual CCM submodule receives the input feature map, $x \in R^{H' \times W' \times C'}$, and applies a DWAC, followed by a DWDC, and then followed by another DWSC, with kernel sizes of 1×3, 3×1, and then 3×3, respectively, and a dilation rate, $d = 2$. Following this set of convolutions is a BN layer and a parametric rectified linear unit (PReLU) activation. The input feature map is then put through another DWAC, DWDC, and DWSC with kernel sizes of 1×5, 5×1, and then 5×5, respectively, and a dilation rate of $d = 3$. Which are then followed by BN and a rectified linear unit 6 (ReLU6) function. The dilation rates aid in the module learning more contextualized local-level features by increasing the receptive field of the convolutional operations. They are also computationally more efficient as they provide a larger amount of coverage with respect to the same number of learnable parameters when compared with standard convolutions. This improves upon the already reduced amount of computational complexity with the choice of separable convolutions in the initial part of the LFE module. The dual CCM submodule can also extract the local features that are eventually integrated to the boundary regions of the polyps through increased filter sizes with dilated convolutions.

As stated previously, each convolution or set of convolutions in the LFE module is followed by a ReLU6, PReLU, or SiLU function as provided by [66]. The mathematics for each of these activations is presented in Equations (3) to (5).

$$\mathrm{SiLU}(x) = x * \sigma(x), \tag{3}$$

$$\mathrm{PReLU}(x) = \max(0, x) + \alpha * \min(0, x), \text{ and} \tag{4}$$

$$\mathrm{ReLU6}(x) = \min(\max(0, x), 6), \tag{5}$$

where x is the input feature map to the nonlinearity, $\sigma(x)$ is the sigmoid function applied to the input feature map, and α is a learnable parameter. The reasons for selecting PReLU are primarily in the fact that it has a learnable parameter and thus can increase model performance by fine-tuning the nonlinearity itself by adjusting its learning rate. The function can also provide an increased performance with respect to accuracy by performing better than ReLU in terms of saturation. Where saturation refers to the condition in CNNs when the output of the nonlinearity approaches the asymptotes of the function, a property that should be avoided to avoid model instability. We select ReLU6 as the second activation in the CCM submodule and provide it at the end of the LFE module. This function is similar to the ReLU activation, however in this module the maximum output value is 6. From our own parametric studies done on the model, we found that maximum performance was reached with the ReLU6 function at the end of the LFE module.

3.2.2 Transformer encoder

The second component of the dual encoder is the transformer encoder, a generic ViT model developed by [62]. This component is depicted in Figure 4. The transformer encoder receives an input image, $x \in R^{H \times W \times 3}$, similar to the PS encoder. The general process for converting the input image to a workable format is as follows: the image is split into patches, the patches are then flattened, and the patches are then converted to a sequence of patch embeddings by linear projection. Learnable parameters are then added to the patch embeddings, which are updated in gradient descent. The embeddings are then fed through the attention mechanism within the transformer encoder, which consists of an MSA module and a multi-layer perceptron (MLP). This attention mechanism is repeated L times before feeding forward the feature map it produces to the partial decoder. The key role of the transformer encoder is to learn global dependencies and

broader semantic context by explicitly learning the relationship between a pixel (or higher-dimensional representation of such a pixel), with respect to all other positions in the feature map. The transformer encoder also employs transfer learning by pretraining the model on the ImageNet dataset before loading the pretrained weights for usage in PSNet.

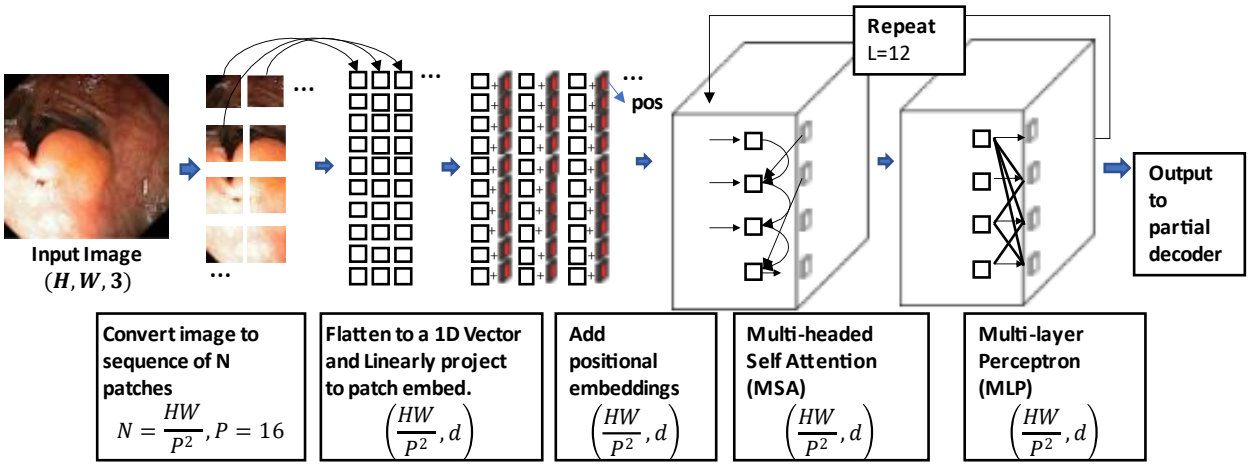

Figure 4. Transformer encoder details [1]

The input image, $x \in R^{H \times W \times 3}$, to the transformer encoder is first converted to a sequence of N patches. The number of patches is a function of the height, H, and the width, W, of the input image, along with the patch size, P. The formula for the number of patches is summarized in Equation (6).

$$N = \frac{H \times W}{P^2} \tag{6}$$

It should be noted that the smaller the patch size, the greater the computational cost. This is because the patch size is inversely proportional to the transformer's sequence length, in which larger sequence lengths require more computational resources. Thus, by decreasing the patch size,

we increase the sequence length. We select a patch size, $P = 16$ to maximize the performance of our transformer encoder and global context extraction, while maintaining an economical computational complexity trade-off. The input image is converted to a sequence of patches and flattened to a 1D vector, as summarized in Equation (7).

$$x = [x_1, \dots, x_N] \in R^{N \times P^2 \times C} \tag{7}$$

The sequence of patches, $x \in R^{N \times P^2 \times C}$, is then linearly projected onto a sequence of patch embeddings, x_0, summarized in Equation (8).

$$x_0 = [Ex_1, \ \dots, Ex_N] \in \ R^{N \times D} \tag{8}$$

where D is the embedding dimension and is a tunable hyperparameter, however it must be linearly proportional to H, W, and P. The embedding dimension, D is kept constant throughout the transformer encoder such that residual skip connections can be employed in the attention mechanism after every block or layer within the attention mechanism. Optimal performance in our network for D was established to be 768, as through parametric testing it was shown to provide a high model accuracy at an acceptable model complexity. Following the creation of patch embeddings, x_0, a set of learnable parameters referred to as the positional embeddings, pos, equivalent in size to the patch embeddings are then added to produce a sequence of tokens, z_o, denoted by Equation (9).

$$z_0 = x_0 + pos \in \ R^{N \times D} \tag{9}$$

The positional embeddings, pos, retain and learn positional information. In the context of image segmentation with transformers, these positional embeddings learn spatial relations between

patches. Following the addition of the positional embeddings, the sequence of tokens is then processed through the attention mechanism in the transformer. The attention mechanism consists of an MSA and an MLP repeated L times. The mathematical formulation of the MSA and the MLP are summarized in Equations (10) and (11), respectively.

$$a_{i-1} = \mathrm{MSA}\big(\mathrm{LN}(z_{i-1})\big) + z_{i-1} \tag{10}$$

$$z_i = \mathrm{MLP}\big(\mathrm{LN}(a_{i-1})\big) + a_{i-1} \tag{11}$$

where $i \in \{1, \ldots, L\}$ and L is the number of layers. L was found through parametric studies to be optimal at $L = 12$. $a_{i-1} \in R^{N\times D}$ is the output of the multiheaded-self attention module, and $z_i \in R^{N\times D}$ is the output of the multi-layer perceptron. Layer normalization (LN) follows each of the MSA and the MLP layers. The mathematics behind the MSA block is formulated in Equation (12) and is depicted visually in Figure 5.

$$\mathrm{MSA}(Q,\ K,\ V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{12}$$

where, $Q \in R^{h\times\frac{HW}{P^2}\times\frac{D}{h}}$ is the queries, $K \in R^{h\times\frac{HW}{P^2}\times\frac{D}{h}}$ is the keys, and $V \in R^{h\times\frac{HW}{P^2}\times\frac{D}{h}}$ is the value. MSA, as described by [67], can be defined as a mapping between the query and a set of key-value pairs to an output. The query, keys, and values are generated by independent filters about the same feature map input. h indicates the number of heads, which is a tunable hyperparameter. The total number of heads applied in the MSA modules within PSNet was $h = 12$. The theoretical understanding of the number of heads is the idea that this operation generates h number of feature extractions, and outlines h different independent features. These features are mapped between the query and key-value pairs.

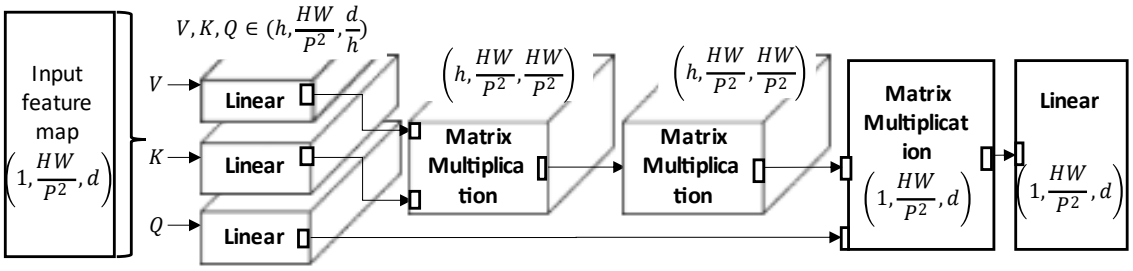

Figure 5. MSA module details [1]

The MLP block in Figure 4 is defined by Equation (13).

$$y = xA^T + b \tag{13}$$

where, y is the output of the MLP, x is the input to the MLP, A^T are the corresponding weights or learnable parameters for the MLP and b is the bias. We apply a pointwise linear layer to the output of the final MLP block, $z_L \in R^{N \times D}$ which then transforms the output signal to $z_{lin} \in R^{N \times 1}$. The MLP uses a gaussian error linear unit (GELU) as its activation function which shows performance enhancements across several image processing tasks when compared to the ReLU function [66].

3.3 Dual decoder

As initially presented in Figure 1, PSNet contains a dual decoder structure. This dual decoder consists of two decoders that work reciprocally by decoding the outputs from their respective encoders, as well as fusing the results with the merge modules that also exist within the dual decoder structure. The major components within the dual decoder consist of the following: the PS decoder, which receives the input feature map from the PS encoder, the partial decoder, which receives the input feature map from the transformer encoder, the enhanced dilated transformer decoder which receives the input feature map from the partial decoder, along with the set of 4

merge modules which receive inputs from the partial decoder and sequential segments of the PS encoder.

The key role of the PS decoder is to efficiently recover local features by decoding the output from the PS encoder to an eventual output segmentation map. The key role of the partial decoder is to convert the output from the transformer encoder from 2D to 3D. The ultimate purpose of the enhanced dilated transformer decoder is to retain the global contextual information received from the transformer encoder, while also extracting even more information in the convolutionally-based operations it feeds the output through. The merge modules have two key roles; 1) they compensate for feature loss by the integration of coarser representations present in the PSNet dual encoder, and 2) they help speed up convergence time with deep supervision.

3.3.1 PS decoder

The PS decoder receives the input feature map, $x \in R^{\frac{H}{16}\times\frac{W}{16}\times 512}$, from the PS encoder, and similar to the PS encoder makes use of a sequence of LFE modules. The PS decoder implements the generic concurrent channel and spatial squeeze-and-excitation (SCSE) module proposed by [53]. The PS decoder also makes use of a newly developed module for the purpose of this research study, the weighted concurrent channel and spatial squeeze-and-excitation module, (wSCSE), a novel modification to the SCSE module. All activations in the PS decoder are SiLU, except for the activations in the SCSE module which are ReLU. The PS decoder is depicted in Figure 6. The final layer in the PS decoder is a PW convolution, which generates an output segmentation map, $x \in R^{H\times W\times 1}$.

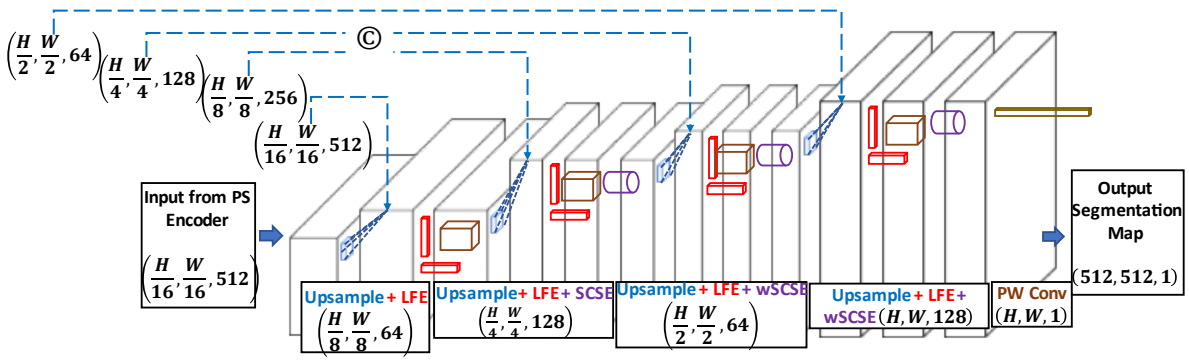

Figure 6. PS decoder details [1]

As seen in Figure 6, the input feature map, $x \in R^{\frac{H}{16}\times\frac{W}{16}\times 512}$ is put through 4 upsampling operations, each followed by an LFE module, in which local feature extraction occurs. Coarse feature maps are also concatenated right before processing through the LFE module through skip connections from the PS encoder which recover global-level feature information that would have otherwise been lost. The dimensions of each of these coarse representations are given in Figure 6 depicted by the dotted blue lines. Following the 2nd, 3rd, and 4th, LFE modules, the input feature map is fed through either an SCSE module or a wSCSE module, as indicated in Figure 6. The primary purpose of these modules is to gather spatial and channel-wise information and mine boundary information present in the input feature maps. The SCSE and wSCSE modules are represented in Figure 7.

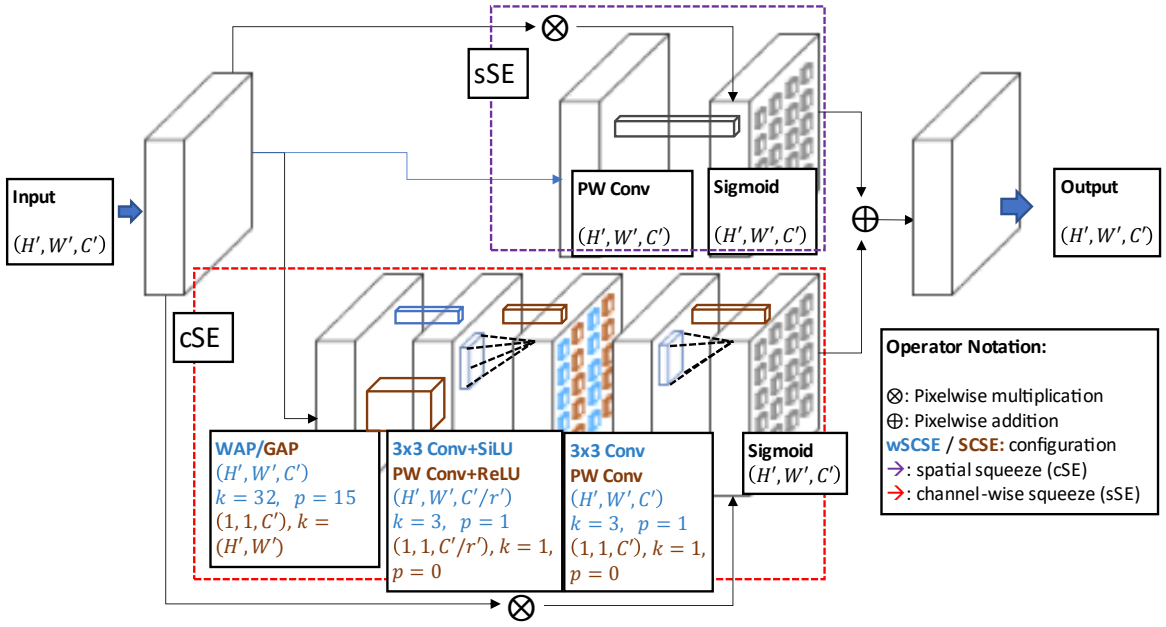

Figure 7. SCSE and wSCSE details [1]

The SCSE and the wSCSE contain two distinct paths, the spatial squeeze component, cSE, and a channel squeeze component, the sSE. The same input, $x \in R^{H' \times W' \times C'}$, is put through both paths. The sSE path is the same for both modules, in which the input is put through a pointwise convolution and a sigmoid activation function. In the cSE path, both the SCSE and the wSCSE module apply a weighting function to the input signal. In the SCSE path this weighting function is global average pooling (GAP). In the wSCSE path this weighting function is weighted average pooling (WAP). The dimensions of the kernels are indicated on Figure 7. The output feature map from these two operations is $x \in R^{1 \times 1 \times C'}$ and $x \in R^{(H' \times W' \times C')}$ for the SCSE and the wSCSE module, respectively. Following the weighting function, the input signal is then put through a 3×3 convolution followed by BN and SiLU for the wSCSE layer, and a pointwise convolution for the SCSE layer followed by BN and ReLU for the SCSE layer. The two aforementioned convolutions and their corresponding dimensions are then repeated and followed by a sigmoid

activation function. The outputs of the sigmoid function from both the sSE and the cSE are added on a pixelwise level of which the output of this operation returns the final output from the (w)SCSE module. This entire operation is formulated in Equation (14).

$$y = x * cSE(x) + x * sSE(x) \tag{14}$$

The SCSE module and the wSCSE module increase the performance of the network by focusing on spatial and channel-wise features, correspondingly reflected in the cSE and the sSE paths respectively. The two models distinguish themselves in the sSE path, where GAP is used in the SCSE module and WAP is used in the wSCSE module. The former results in unit dimensions with respect to the height and width of the feature map, and are kept constant throughout the sSE path. Whereas, the latter results in the dimensions with respect to height and width being kept constant throughout its respective sSE path. The advantage of GAP usage in the SCSE module is the lower computational complexity, while still maintaining important spatial feature information by highlighting the most important features. Whereas, the advantage of the WAP is that it highlights boundary pixel information with its specific configuration of the kernel and stride (which consequently give it its constant height and width dimensions throughout the sSE path). The WAP weighting function assists in mining boundary cues. This idea was taken from the loss function found in [55]. The one disadvantage of the wSCSE module is that because of the WAP layer, the dimensions are kept constant throughout the sSE path which results in a longer runtime relative to the SCSE module. However, the benefits are found in the increased boundary pixel identification and the increased focus on more complex characteristics in the input feature map. This was also evidenced through parametric studies in which the wSCSE module clearly improved performance in two locations, which are shown in Figure 6 and are included in the final version of

this model. Moreover, despite the trade-offs with respect to module runtime, the number of parameters is the same in both the wSCSE and SCSE module.

3.3.2 Partial decoder

The partial decoder receives the input from the transformer encoder, $x \in R^{N \times d}$, and converts this 2D input to a 3D output, $x \in R^{\frac{H}{16} \times \frac{W}{16} \times \mu}$, where μ is the number of logits. The partial decoder is generic and was originally designed by [68]. The partial decoder is presented in Figure 8.

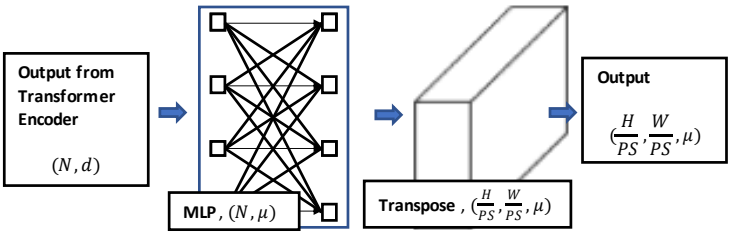

Figure 8. Partial decoder details [1]

As can be seen in Figure 8, the input is transformed to a set of logits using a standard MLP. The output of this MLP is $x \in R^{N \times \mu}$. We treat the number of logits, μ, as a hyperparameter. Where we treat each logit within the set to be a specific type of pixel, such as boundary pixels, pixels detailing regular occurring regions of obscurity, pixels detailing the boundary of the camera lens, etc. Through parametric studies, it was found that the maximum performance associated with the number of logits was $\mu = 64$. Good performance was also shown at $\mu = 32$, and reasonable performance was also found at $\mu = 1$ (i.e. classification as simply either healthy tissue or polyp). The final operation in the partial decoder is the rearrange or transpose function. This function reorganizes the relative positions of the output from the MLP to 3D. Using the rearrange function from the einops library, the output is reshaped to a feature map, $x \in R^{\frac{H}{16} \times \frac{W}{16} \times \mu}$, which is then

administered to the enhanced dilated transformer decoder, as indicated in Figure 1. This feature map can be described as a coarse representation of the candidate output segmentation map that will be generated by the transformer component of PSNet. The output from this layer is also bilinearly upsampled to feature maps $x^{1/2} \in R^{\frac{H}{2}\times\frac{W}{2}\times 64}$, $x^{1/4} \in R^{\frac{H}{4}\times\frac{W}{4}\times 64}$, and $x^{1/8} \in R^{\frac{H}{8}\times\frac{W}{8}\times 64}$ for input into the merge modules.

3.3.3 Enhanced dilated transformer decoder

The enhanced dilated transformer decoder receives the input feature map from the partial decoder, $x \in R^{\frac{H}{16}\times\frac{W}{16}\times\mu}$, and then puts the input through a sequence of two upsampling operations, which is then followed by a standard convolution and then a dilated convolution. Each convolution uses a 3×3 kernel, followed by BN and SiLU activations. The dilation rate of the dilated convolutional layer is $d = 3$. The primary reason behind selecting the dilated convolutions for this layer was to further enhance the global feature extraction with the larger receptive field, without loss of resolution. Thus, improving upon the key role of the transformer component of PSNet further. Even though dilated convolutions have potential to lose feature information [65], when running our own parametric studies in designing this decoder, it was found that the single dilated convolutional layer improved performance of the model when compared to an entirely standard convolution implementation of the decoder. Following the upsampling and convolutional component in the first half of the decoder, two pointwise convolutions are implemented, as seen in Figure 9, detailing the overall schematic of the enhanced dilated transformer decoder.

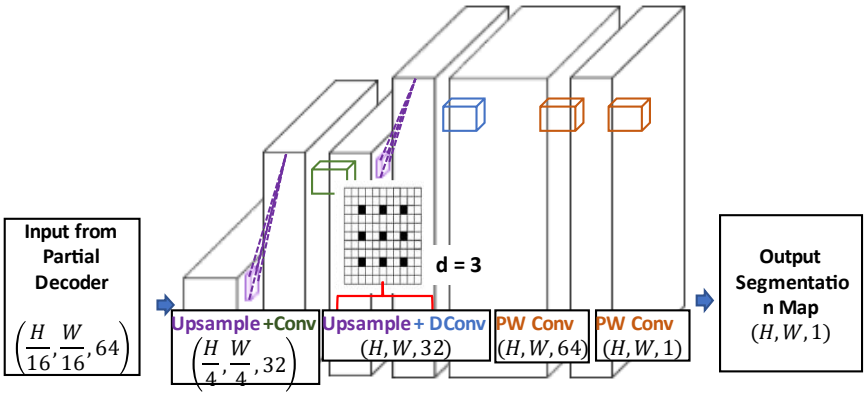

Figure 9. Enhanced dilated transformer decoder details [1]

3.3.4 Merge module

In order to integrate the coarse representations of the transformer encoder and the PS encoder, the merge module was designed and deployed at different feature scales throughout the decoder. The merge module amalgamates the two feature maps from the PS encoder, $x_{PS} \in R^{\frac{H}{R} \times \frac{W}{R} \times C_i}$, and indirectly the transformer encoder via the partial decoder, $x_t \in R^{\frac{H}{R} \times \frac{W}{R} \times C_i}$, where $R \in \{2, 4, 8, 16\}$ and corresponds to the feature scales of $1 / 2$, $1 / 4$, $1 / 8$, and $1 / 16^{th}$ the original input size, respectively. The merge module amalgamates these tensors via concatenation along the channel dimension. A diagram detailing the structure of the merge module is presented in Figure 10.

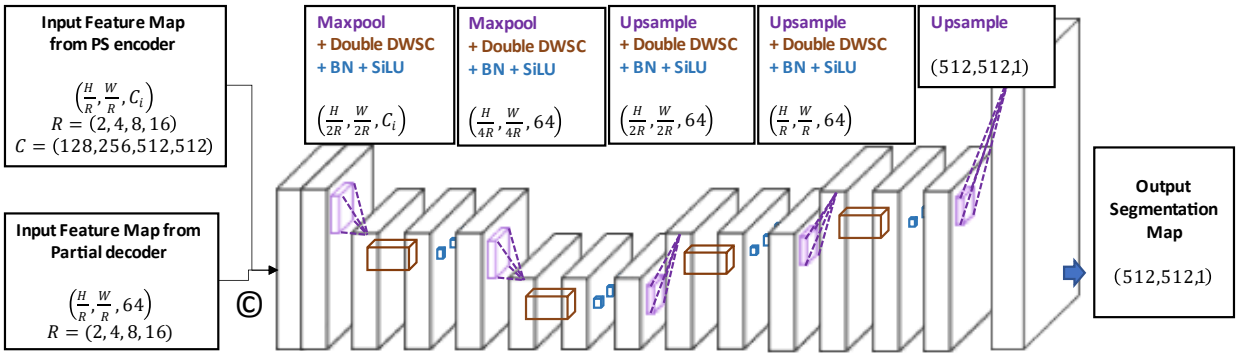

Figure 10. Merge module details [1]

The merge module takes in input feature maps from the PS encoder and the bilinearly upsampled output of the partial decoder, in which the latter is upsampled to the dimensions of the input from the PS encoder. The two features, the input from the PS encoder and the upsampled output from the partial decoder, are then concatenated to produce a tensor, and the output is then put through a sequence of two maxpooling layers, each followed by a 3×3 DWSC with BN and SiLU. Following the progressive downsampling, the feature map is then put through two upsampling operations, each followed by 3×3 DWSC with BN and SiLU. Finally, the feature map is upsampled to present a candidate output segmentation map, $x \in R^{H \times W \times 1}$. The number of channels is kept constant at 64 throughout the entire merge module to reduce computational resource extraction.

3.4 Loss function

Over the past several years, there has been a growing trend amongst researchers to generate new loss functions capable of increased boundary detection, as seen in models developed by [55], [45], and [58]. In each of these aforementioned models, the new approach is to combine an adaptive average pooling weighting layer, like the WAP layer presented in the wSCSE module, with binary cross entropy (BCE). This feature map then proceeds into an IoU loss function. The purpose of this layer progression is to focus on boundary pixels. We show, by virtue of our SOTA results, that this overcomplication isn't needed. In its place we use a basic IoU loss function, also known as the Jaccard index [69]. This loss function is summarized in Equation (15), along with our mathematical formulation of the output feature map from the model summarized in Equation (16).

$$L_n = \frac{|\sigma(\overline{x_n}) \cap y_n|}{|\sigma(\overline{x_n}) \cup y_n|} \tag{15}$$

$$\overline{x_n} = \frac{x_n{}^{T} + x_n{}^{C} + x_n{}^{1/16} + x_n{}^{1/8} + x_n{}^{1/4} + x_n{}^{1/2}}{6} \tag{16}$$

where, x_n is the model architecture's output and has the same dimensions as the input image to the model, except only one channel dimension. σ is the sigmoid function and $\sigma(x_n)$ denotes the sigmoid activation of the output from the model architecture which is the output segmentation map. $x_n{}^{T}$ and $x_n{}^{C}$ refer to the outputs from the enhanced dilated transformer decoder and the outputs from the PS decoder, respectively. $x_n{}^{1/16}$, $x_n{}^{1/8}$, $x_n{}^{1/4}$, and $x_n{}^{1/2}$ refer to the outputs from the merge modules obtained from the original corresponding scales indicated by the superscript. These variables constitute the deep supervision component of the loss function. The output segmentation map produced by the sigmoid activation of the output from the model is essentially a 3D tensor of probabilities, with each pixel representing a probability that it belongs to the particular class the model is trying to learn. In our study, this would be representing the likelihood that each pixel corresponds to either healthy tissue or a polyp. A score closer to 1.0 would indicate that the pixel is more likely to be a polyp and a score closer to 0.0 would indicate that the pixel is more likely to be healthy tissue. We also denote y_n as the ground truth for the model and n refers to the individual pixel ranging in value from 0 to N, where N is the total number of pixels.

Being one of the more common evaluation metrics selected for image segmentation tasks, the primary reason behind selecting the IoU metric is its demonstrated ability in segmentation tasks and its property of scale invariance. Scale invariance refers to the lack of change in scales (width, height, and channel) when multiplied by a common factor, and that all these dimensions are considered in computing the IoU score. In the context of our segmentation task with polyps ranging

in a wide variety of sizes, this metric is an excellent choice as the IoU metric then focuses on the total area of overlap between the output segmentation map and the ground truth [70].

Chapter 4. Training and Implementation Details

Deep learning-based methodologies are evaluated by benchmarking the performance of a model on standardized datasets. While there is no official standard associated with polyp segmentation and deep learning-based methodologies used to accomplish this task, there exists a number of datasets that frequently appear in polyp segmentation papers, such as in [45], [34], [55], [57], and [58]. Thus, in this chapter, a review of these frequently used datasets used to evaluate a polyp segmentation model's performance and generalization capabilities is provided, followed by a detailed discussion of the features of the parameters and modules used to train the network.

4.1 Datasets

Review of current literature determined that the number of open-source datasets of polyps for general public use is limited. Therefore, only 5 publicly available datasets were evaluated for the purpose of this research project. The datasets used in the training and validation of the model were the Kvasir-SEG [13], the CVC-ClinicDB [14], the CVC-ColonDB [15], ETIS [16], and the EndoScene [17] datasets. Each dataset consists of 2 subsets, the input images and the ground truths. The ground truths are a binary grayscale map with each pixel corresponding to a 1 or a 0, where 1 indicates a polyp and 0 indicates healthy tissue or non-polyp.

Division of datasets and training procedures followed that of other SOTA models such as [45], [55], and [58]. The procedure that these models followed was replicated in this research study in an effort to report comparable model performance metrics. This method combined and split the Kvasir-SEG and CVC-ClinicDB dataset into training and test sets, and used the CVC-ColonDB, ETIS, and EndoScene datasets exclusively as test sets. It should be noted that the EndoScene dataset is a combination of the CVC-ClinicDB and the CVC-300 set. Given that we already include

the CVC-ClinicDB dataset in our procedure, we exclude the overlap with this set and just use the data from the CVC-300 set, as described in [55].

The total training set size is then 1,450 images combined from the Kvasir-SEG and the CVC-ClinicDB dataset. The remaining 10% from these sets are also used for individual test sets used in evaluation of the model performance. The validation set was created by taking 10% of the training set. The dataset divisions from each of their respective parent sets is summarized in Table 1. A visual showing a sample of some of the polyps from each dataset is shown in Figure 11.

Table 1. Datasets used in case studies [1]

Dataset	Resolution	# of total images	# of train images	# of validation images	# of test images
Kvasir-SEG [13]	487 × 332 to 1920 × 1072	1,000	900	90*	100
CVC-ClinicDB [14]	384 × 288	612	550	55*	62
CVC-ColonDB [15]	574 × 500	380	-	-	380
ETIS [16]	1225 × 966	196	-	-	196
EndoScene [17]	574 × 500	60	-	-	60

Figure 11. Image and ground truth data samples [1]

The Kvasir-SEG, CVC-ClinicDB, CVC-ColonDB, ETIS, and EndoScene datasets contain a wide variety of polyps, with each set containing an assortment of polyps with different relative sizes to the camera view, along with different colors and textures present in the images. Each dataset provides a mask or ground truth detailing on a pixel-based level, the location of polyps. The Kvasir-SEG dataset contains 1,000 variable sized images of polyps. The resolution of the images varies between 487×332 and 1920×1072. The CVC-ClinicDB dataset contains 612 images of polyps of resolution size 384×288. This dataset was generated from still images from 31 colonoscopy videos. The CVC-ColonDB dataset contains 380 images of polyps of resolution size 574×500, with each image coming from 15 distinct colonoscopy sequences. The ETIS dataset contains 196 images of polyps of resolution size 1225×966. The EndoScene dataset contains 60 images of polyps of resolution size 574×500.

4.2 Training and optimization

The model was built in the PyTorch framework in the Python computing language. The model was trained on 4 NVIDIA A100 GPUs, using multi-GPU training. Image preprocessing was applied to the images and was relatively simple, in the sense that the only transformations applied were image

resizing and then normalization. The images were resized from their original resolutions, as specified in Table 1, to an image size of 512×512. Following image resizing, the images were then normalized with a mean, μ, and a standard deviation, σ, of 0.5 and 0.5, respectively.

The model was updated using SGD, an optimization method used to update the learnable parameters within a CNN. For an arbitrary learnable parameter, such as the weights present in the convolutional, linear, or PReLU layers, this mechanism is characterized as follows:

$$w_{i+1} = w_i - \alpha \frac{dL}{dw_i} \tag{17}$$

where, w_i is the value of the learnable parameter at epoch i, w_{i+1} is the value of the learnable parameter used for the next epoch, α is the value of the learning rate, $\frac{dL}{dw_i}$ is the derivative of the loss function with respect to the learnable parameter at epoch i. We define an epoch to be a full completion of the model iterating through the entire dataset. The Adam optimizer was used to further modify the weight update in SGD, in which the parameters for this optimizer are presented in Table 2, along with details pertaining to training such as the loss function, total number of epochs, and the batch size.

Table 2. Optimizer and training hyperparameters [1]

Optimization	Learning rate scheduler	Learning rate	β_1	β_2	# of epochs	Loss function	Batch size
Adam	Warmup poly	2.1052×10^{-5}	0.9	0.999	400	IoU	16

Weights were initialized variably in layers, depending on the type of layer. To avoid convergence issues and model instability during training, transfer learning was implemented and pretrained weights were used for the transformer encoder. The weights were trained on the ImageNet21k dataset [62]. For the convolutional layers, the weights were initialized from a normal distribution with a $\mu = 0$ and $\sigma = 0.02$. These values were also used for the initialization of the linear or fully connected layers. For the LN and BN layers, the weights and biases were initialized to 1.0 and 0.0 respectively. As shown in Table 2, the learning rate was initialized at 2.1052×10^{-5} and then decreased with a polynomial learning rate scheduler after an initial warmup period. This is governed by the equation below.

$$\alpha_{i+1} = \begin{cases} \alpha_i \left(\frac{\bar{e}_i + 1}{r}\right)^{0.98}, & x < \bar{e}_t \times r \\ \alpha_i \left(1 - \frac{\bar{e}_i - r}{\bar{e}_t - \bar{e}_t \times r}\right)^{0.98}, & x \geq \bar{e}_t \times r \end{cases}$$

where, $\bar{e}_i$ is the current epoch, $\bar{e}_t$ is the total number of epochs r is the warmup ratio, α_i is the initial value of the learning rate, and α_{i+1} is the value of the learning rate of the next epoch. After iterating through the total number of epochs, the filters are learned in all layers with learnable parameters and the model is capable of segmentation of polyps.

4.2.1 Additional model performance metrics

In addition to the IoU loss function used for model training and performance quantifying, as shown previously in Equation (15), a number of other model performance metrics were evaluated throughout training and testing. Similar to the IoU function, the Dice loss [71] was calculated. In addition to these metrics, the true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) were also calculated, which were used to calculate the Precision, Recall and Accuracy. These measures are summarized in Equations (18) to (21) below.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{18}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{19}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{20}$$

$$\text{L}(\text{x}, \text{y}) = \text{L}_{\text{dice}} = \frac{2 \times |\sigma(\text{x}_\text{n}) \cap \text{y}_\text{n}|}{|\sigma(\text{x}_\text{n}) + \text{y}_\text{n}|} \tag{21}$$

where x_n is the output from the model architecture, σ is the sigmoid function and $\sigma(x_n)$ denotes the sigmoid activation of the output from the model architecture which is the output segmentation map. y_n is the ground truth for the model and n refers to the individual pixel ranging in value from 0 to N, where N is the total number of pixels.

Chapter 5. Ablation Studies

A number of case studies were conducted that evaluated several components of the model and their quantifiable contribution to its ability to segment images. Moreover, this study implicitly studied the model's computational complexity and the effect of increasing the number of parameters on the model's overall performance. The model was evaluated without the following components previously discussed: the dual decoder, the merge modules, the PS encoder, and without CCM submodules in any of the components in the PS encoder or PS decoder. As expected, the computational complexity of the model increased in a logarithmic fashion, clearly indicating the tradeoff between computational complexity and performance, as shown in Table 3.

Table 3. Case studies [1]

Configuration	Kvasir		CVC-ClinicDB	
	mDice	mIoU	mDice	mIoU
PSNet (- dual decoder)	0.817	0.729	0.867	0.787
PSNet (- merge modules)	0.876	0.815	0.890	0.839
PSNet (- PS encoder)	0.908	0.854	0.908	0.864
PSNet (- CCM)	0.916	0.865	0.916	0.867
PSNet	0.929	0.879	0.928	0.879

To measure the performance of the dual decoder, we substitute the dual decoder with bilinear upsampling. For both the PS encoder and the partial decoder, the outputs are merely pointwise convolved to get a dimensionality transform to a channel dimension equal to 1, and then

bilinearly upsampled with a scale factor of 16. The removal of the dual decoder has a profound effect on the accuracy of the model. By inference we can determine that for the Kvasir-SEG dataset, the introduction of the dual decoder improved the accuracy of the model by 11.2% for the mDice score and 15% for the mIoU score. For the CVC-ClinicDB dataset, the introduction of the dual decoder improved the accuracy by 6.1% and 9.2% for the mDice and mIoU scores, respectively. We attribute this performance increase due to the additional feature extraction that occurs in the dual decoders, both in the individual layers and the skip connections in the PS decoder, along with the local feature extraction in the enhanced dilated transformer decoder.

For the evaluation of the merge modules, we simply remove them from the network and just average the candidate output segmentation maps produced by the transformer branch and the PS encoder-decoder branch. We observe a solid increase in the performance due to the introduction of the merge modules. For the Kvasir-SEG dataset we observe a 5.3% and a 6.4% increase in the mDice and mIoU scores, respectively. For the CVC-ClinicDB dataset we observe a 3.8% and a 4% increase in the mDice and mIoU scores, respectively. Thus, we can clearly identify the importance of the additional candidate segmentation maps in improving the accuracy of the model. While this was not the intention of introducing deep supervision to the network, as these modules were originally put into the network to speed up convergence times, this was an added benefit of the modules. We attribute this not however to the presence of deep supervision, but rather just to the increased number of parameters and the reintroduction of coarse information that would have been lost without the modules.

For the evaluation of the PS encoder, we remove the PS encoder and use the output from the transformer encoder as input to the PS decoder. We observe a smaller increase in the performance of the model due to the PS encoder. For the Kvasir-SEG dataset we observe a 2.1%

and 2.5% increase in the mDice and mIoU scores, respectively. For the CVC-ClinicDB dataset we observe a 2% and 1.5% increase in the mDice and mIoU scores, respectively. The lower reduction in performance due to the presence of the PS decoder makes sense intuitively since the transformer encoder is the heavier component of the dual encoder in terms of number of parameters, and that global context is generally more valuable than local features. However, we determine that the introduction of the PS encoder and its local feature extraction capabilities are paramount in distinguishing PSNet as SOTA.

Finally, for the evaluation of the CCM submodules. Each CCM component was removed from the PS encoder and PS decoder. The smallest decrease in model performance is observed with the removal of this component. For the Kvasir-SEG dataset we observe a 1.3% and 1.4% increase in the mDice and mIoU scores, respectively. For the CVC-ClinicDB dataset we observe a 1.2% and 1.2% increase in the mDice and mIoU scores, respectively. Again, this makes sense intuitively due to the nature of the relative contribution of these components with respect to the total number of parameters. However, a single percent increase is significant when it comes to establishing SOTA results, and thus we demonstrate the effectiveness of these modules in improving the performance of PSNet.

5.1 Comparative studies

To evaluate the model performance and determine how our model compared to recent SOTA models in polyp segmentation, 5 major models were studied and evaluated against. These models were UNet [32], UNet++ [60], PraNet [55], AMNet [58], and TransFuse [45]. The majority of these models feature an encoder-decoder based structure, two methods employ a parallel partial decoder and the final method falls under the category of an ensemble method, similar to ours.

The results for the comparative studies are summarized in Table 4. In Figure 12 we provide a sample of the outputs with their corresponding ground truths and image inputs.

Table 4. Evaluation scores of PSNet vs. SOTA models [1]

	Kvasir		ClinicDB		ColonDB		EndoScene		ETIS		Average	
	mDice	mIoU	mDice	mIoU	mDice	mIoU	mDice	mIoU	mDice	mIoU	mDice	mIoU
UNet	0.818	0.746	0.823	0.755	0.512	0.444	0.710	0.627	0.398	0.335	0.652	0.581
UNet++	0.821	0.743	0.794	0.729	0.483	0.410	0.707	0.624	0.401	0.344	0.641	0.570
PraNet	0.898	0.840	0.899	0.849	0.709	0.640	0.871	0.797	0.628	0.567	0.801	0.739
AMNet	0.912	0.865	0.936	0.888	0.762	0.690	---	---	0.756	0.679	0.842	0.781
TransFuse	0.92	0.87	0.942	0.897	0.781	0.706	0.894	0.826	0.737	0.663	0.855	0.792
Ours (PSNet)	0.929	0.879	0.928	0.879	0.795	0.715	0.877	0.802	0.787	0.713	**0.863**	**0.797**

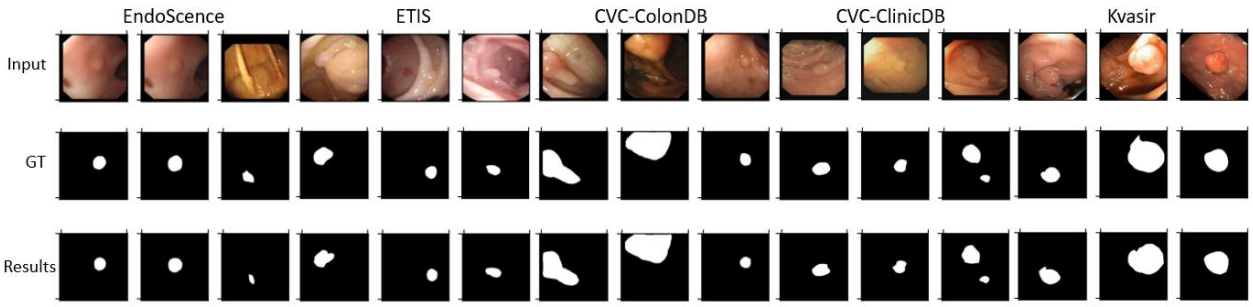

Figure 12. Sample model outputs vs. input images and ground truths [1]

It can be observed from Table 4 that the best performing datasets were the datasets corresponding to the training data, Kvasir-SEG and CVC-ClinicDB. Whereas, the worst performing dataset was the ETIS dataset, followed by the CVC-ColonDB dataset and then followed by the EndoScene dataset. We attribute the EndoScene dataset as performing reasonably well given that it's one of the smallest sets and the images generated from this dataset were similar

to the CVC-ClinicDB dataset. It can be deduced that ETIS performs the worst primarily because the polyps associated with this dataset are smaller on average and show the widest variation in size and color relative to the Kvasir-SEG and CVC-ClinicDB datasets. Broadly speaking, the model classified larger polyps much better than smaller polyps, which is likely because larger polyps contain more surface area of polyp relative to boundary pixel, which produces a higher accuracy if the majority is calculated correctly. For smaller polyps, the opposite is true. However, ultimately in simple machine learning terms, we ascribe the issue of better performance on the Kvasir-SEG and the CVC-ClinicDB dataset to the fact that the model was trained on and observed only data from the two aforementioned sets. Put more explicitly, the dataset was *biased* towards the Kvasir-SEG and CVC-ClinicDB datasets. Thus, through Table 4, we show that PSNet significantly outperforms other models with respect to the ETIS dataset, and overall our model shows improvement with respect to the cumulative mDice and mIoU score across all datasets, thus demonstrating SOTA performance.

The convergence value that the loss moved towards was approximately 0.79, with a best loss of 0.797. The best results were obtained at epoch 218 over the course of the total 400 epochs of training. We summarize our loss and other performance metrics at this epoch in Table 5. Also, in an effort to visualize the model training and describe the general behavior of the model throughout each epoch, a loss curve describing the training of the model with the original configuration of the dataset is provided in Figure 13.

Table 5. Summary of performance metrics for original dataset [1]

Metric	mDice	mIoU	Accuracy	Precision	Recall	Params
-	0.863	0.797	0.984	0.870	0.895	90M

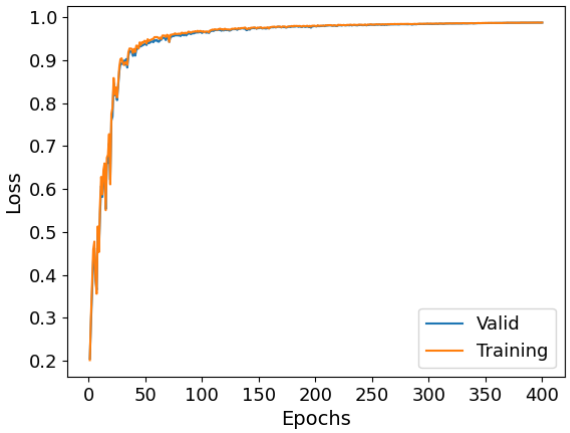

Figure 13. Loss curve - original dataset

The precision and recall reported for the best epoch during model training was 0.870 and 0.895, respectively, as seen in Table 5. The Recall value is higher than the Precision score. Referencing Equation (18) which describes the mathematical formulation of Precision, a higher precision indicates that the model has a tendency to return pertinent values. This can be inferred from the fact that it takes the ratio of the number of true positives, to the total number of positives (both true and false). In the context of our polyp segmentation problem, this has the physical meaning of how well the model classifies *polyp tissue*, disregarding how well the model evaluates healthy tissue. Equation (19) describes the mathematical formulation for Recall. The formula for Recall similarly indicates the importance of relevant values, i.e., true positives, however this formula also incorporates the number of false negatives, which is an important parameter in the context of polyp segmentation. This is because the returning of false negatives may have serious consequences for patient outcomes. Furthermore, when the field-of-view in the camera is imaging small polyps, the number of false negative becomes even more crucial, as the ability to capture small polyps accurately is necessary for clinical viability. Therefore, it is clear that the Recall score

being relatively higher than the Precision score is actually valuable in this context of pixelwise polyp detection.

As can be seen in Figure 13, convergence was reached by approximately 100 epochs. The validation and training sets are nearly identical, which makes sense to the degree that the validation set is data that the training set has already seen before. Mild oscillation is also observed in the approach towards equilibrium in training. This could be addressed and ameliorated by a different learning rate scheduler or different parameters chosen in the Adam optimizer. However, the oscillation isn't a big deal given that the model achieves SOTA performance.

Another issue is the overall smaller number of images used for training at 1,450 images total. One recommendation to resolve this overfitting is to increase the size of the training set with more images of polyps. This could be done through data augmentation methods, such as cropping or color modifications, or through alternative deep learning approaches, such as with a generative adversarial network (GAN). Another method would be to actually gather more images through more colonoscopy procedures.

We implement our own solution to this problem by creating a new dataset which amalgamates all the images from the 5 publicly available datasets covered in this thesis. The new dataset combines all 5 datasets and shuffles them thoroughly into a total set of 2,248 images, which were then divided into a train, test, and validation set. We summarize the set divisions of this new dataset in Table 6, and present the results associated with this set in Table 7.

Table 6. New merged dataset breakdown [1]

Set	Train	Valid	Test

Number in dataset	2023	224	224

Table 7. New merged dataset result statistics [1]

	Average	
	mDice	mIoU
Ours (PSNet)	**0.941**	**0.897**

The results for training the model with the new merged set are presented in the loss curve shown in Figure 14.

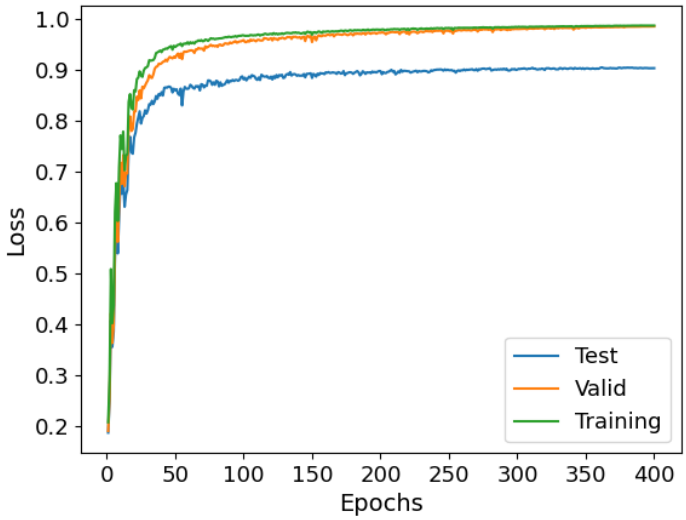

Figure 14. Loss curve summary for merged dataset

It can be seen that in comparison to Figure 13, with the original dataset, that convergence is similar, as seen with the new dataset in Figure 14. The maximum loss achieved is higher, at an equilibrium value of approximately 0.89 for the test set. Despite shuffling the data thoroughly prior to division between test, validation, and train sets, there remains a gap between the results of the

test set and the training and validation sets. This is due to the fact that the new dataset, like the original, is overall imbalanced and is an issue that is ultimately unresolvable. To resolve this issue would require an equal representation from every dataset used to generate the new master merged dataset, i.e., Kvasir-SEG, CVC-ClinicDB, CVC-ColonDB, ETIS, and the EndoScene dataset. While there remains a gap between the test set and validation and training set, the two figures converge with the same shape and are much closer together, thus demonstrating less overfitting. Moreover, the validation and the training set also exhibit a noticeable gap in fit, thus further demonstrating the benefit of the new merged set.

Chapter 6. Conclusions and Future Work

This research project attempted to generate a polyp segmentation network. The results of this attempt were referred to as PSNet. PSNet is capable of real-time polyp segmentation and provides SOTA results with respect to this task, evidenced by its high mIoU and mDice score, and therefore accomplishes the original goals of the project. In this chapter we provide a summary of the research methods including the overall structure of PSNet and the technical contributions of PSNet, along with how SOTA performance was established, as well as a discussion of limitations of the proposed PSNet, and future work within the field of polyp segmentation as a consequence of this research study.

6.1 Conclusion

The technical contributions of PSNet consists of its SOTA performance along with the newly developed components within the dual encoder-decoder structure that work to provide its SOTA performance. These components are as follows: the PS encoder, the PS decoder, the merge modules, and the enhanced dilated transformer decoder. The model also consists of two generic components, the transformer encoder and the partial decoder. Together these components provide a novel dual model architecture capable of polyp segmentation in real-time. It is hoped that some of these newly developed modules will be used in other research studies of a similar nature within the field of medical image segmentation.

The model works reciprocally by feeding an input image synchronously through the PS encoder and the transformer decoder, which then proceed respectively through the PS decoder and then the joint partial decoder and enhanced dilated transformer decoder. These two components generate two candidate segmentation maps, one from the PS encoder-decoder structure and one from the transformer. Intermediate feature maps are put through the merge modules to incorporate

deep supervision and additional output segmentation maps are generated from the merge modules from 4 scales. All 6 segmentation output maps are then averaged to produce the final segmentation output map. Skip connections from the PS encoder-decoder and the attention mechanisms inherent to the transformer are implemented to further learn broader contextual information.

PSNet was trained using SGD, of which the optimization method was further modified with the Adam optimizer and a warm-up polynomial learning rate scheduler to further assist in convergence of the model during training, as well as reduce numerical instability by ultimately reducing the learning rate gradually. The batch size used was 16 and the model was trained on 4 NVIDIA-A100 GPUs, using multi-GPU training. The model itself was built in the PyTorch framework using the Python computing language. The model was trained on 5 publicly available datasets. The number of training images in the training set was 1,450. The number of images in the test set was 100, 62, 380, 196, and 60 for the Kvasir-SEG, CVC-ClinicDB, CVC-ColonDB, ETIS, and EndoScene sets, respectively. SOTA performance was achieved with respect to the combined mIoU and mDice score, which were correspondingly averaged from the individual mIoU and mDice scores evaluated on each of the 5 test sets.

The ultimate goal of this research study was to address several extant issues within the field of polyp segmentation. Issues associated with polyp segmentation are accurate identification of boundary pixels, as well as overfitting and model generalization. These issues were clearly addressed with the introduction of PSNet, as SOTA performance was achieved, as well as with the introduction of the new merged dataset which further reduces overfitting across all datasets. The final conclusions for the research study of polyp segmentation are as follows:

- Using both the global feature scale advantages of transformers and the local feature scale advantages of CNNs, we provide a novel amalgamation of the two approaches and

achieve SOTA performance with relatively lower computational complexity compared to other dual models.

- Two methods with respect to dataset configuration were used for evaluating SOTA performance and addressing the design issues of boundary pixel identification and improving model generalization. The first method was to use a standardized dataset, that combined both the Kvasir-SEG and the CVC-ClinicDB dataset, while reporting results on unseen data from the 5 major publicly available datasets, Kvasir-SEG, CVC-ClinicDB, CVC-ColonDB, ETIS, and the EndoScene datasets. The second method was to create a new merged dataset from these aforementioned sets and create a new simple training, validation, and test set division that improved the model's ability to generalize.
- Boundary pixel definitions were improved upon by the usage of the transformer encoder and its configuration with the dual encoder. The transformer encoder was able to focus more on global context and therefore boundary pixels by offloading some of the work it would have to do independently with local feature extraction by simultaneous implementation with the PS encoder. Boundary pixel definitions were also improved upon with the wSCSE module, the skip connections between the PS encoder and the PS decoder, along with the dilated convolutions in the LFE modules throughout the PS encoder and the PS decoder. These novel modifications also improved upon the model generalization capabilities.
- For the original, standardized training and test sets, PSNet reports a combined mDice and mIoU of 0.863 and 0.797, respectively.
- For the new, merged dataset, PSNet reports an mDice and mIoU of 0.941 and 0.897, respectively.

- Using the original, standardized training and test sets, PSNet improved upon SOTA performance by the highest performing SOTA model, TransFuse, by 1% with respect to both mIoU and mDice across all 5 publicly available datasets. With respect to the new merged dataset, PSNet improved upon this mDice score by 8.6% and the mIoU score by 10.5%.
- Using the original, standardized training and test sets, PSNet improved upon SOTA performance by the most recently published SOTA model, AMNet, by 1.6% and 2.1%, respectively, for the mIoU and mDice across all 5 publicly available datasets. With respect to the new merged dataset, PSNet improved upon this mDice score by 9.9% and the mIoU score by 11.6%.

6.2 Limitations

A limitation of the proposed research method is the dataset preparation. This is seen in the results of the standardized original set. However, it is also observed in the new merged set, as the total number of images still remained constant, however overfitting was slightly reduced with the second dataset configuration. In both cases, the number of images in the train set was low, ≤ 2000 images of polyps for each. Thus, the model's generalization capabilities could be explored further by incorporating new data. As stated previously, this task could be accomplished through generation of new images with data augmentations or with GANs. New endoscopic images could be generated in clinical settings as well. Moreover, beyond just generalization for polyp segmentation, no inferences have been made on the model's ability to segment other forms of medical images such as retinal images, images of the brain, etc.

Another limitation of the research method is that PSNet contains a large number of parameters at 90M. PSNet was designed to incorporate a minimum number of parameters through

the careful integration of a variety of economical modules such as SCSE modules and separable convolutions. However, the fact that PSNet is an ensemble method and employs the usage of both transformers and CNNs results in the larger computational complexity of the model. That being said, relative to TransFuse, the highest performing SOTA model presented in [45], the number of parameters is significantly less with 14% less parameters in total, while structured with a combinatory transformer and CNN dual model approach.

6.3 Future work

In future work, an increased effort to reduce overfitting is necessary. Current SOTA models use a standardized method that involves evaluation on a dataset that is extremely limited and inherently prone to overfitting and generalization issues. This issue could be addressed by the collection of more data from endoscopic images. In doing this, a new standardized dataset could be proposed. Hardware applications of PSNet could also be explored, in which a camera used in colonoscopy procedures could be overlayed digitally on to the video file that would then mask or identify the location of the polyp on a pixel-based level.

References

[1] J. Lewis and Y. J. Cha, "Dual Encoder-Decoder-based Deep Polyp Segmentation Network for Colonoscopy Images," *IEEE Transactions on Medical Imaging,* vol. In Review, 2022.

[2] M. S. Nixon, A. S. Aguado and M. S. Nixon, *Feature extraction & image processing for computer vision*, Oxford: Elsevier Science & Technology, 2012.

[3] National Institute of Health, *Magnetic resonance imaging*, Bethesda, MD: U.S. Dept. of Health and Human Services, 1988.

[4] P. Grangeat, *Tomography*, Wiley: London, 2009.

[5] G. S. Lodwick, "Computer-aided Diagnosis in Radiology: A Research Plan," *Investigative radiology,* vol. 1, no. 1, pp. 72-80, 1966.

[6] D. D. Boo, M. Prokop, M. Uffmann, B. v. Ginneken and C. Schaefer-Prokop, "Computer-aided detection (CAD) of lung nodules and small tumours on chest radiographs," *European journal of radiology*, vol. 72, pp. 218-225, 2009.

[7] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, Cambridge: MIT Press, 2016.

[8] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-Based Learning Applied to Document Recognition,” in *Proceedings of the IEEE*, 1998, vol. 86, no. 11, pp. 2278-2324.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097-1105.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International journal of computer vision,* vol. 115, no. 3, pp. 211-252, 2015.

[11] R. Zhang, Y. Zheng, T. W. C. Mak, R. Yu, S. H. Wong, J. Y. W. Lau and C. C. Y. Poon, "Automatic Detection and Classification of Colorectal Polyps by Transferring Low-Level CNN Features From Nonmedical Domain," *IEEE journal of biomedical and health informatics,* vol. 21, no. 1, pp. 41-47, 2017.

[12] P. Ngoc Lan, N. S. An, D. V. Hang, D. V. Long, T. Q. Trung, N. T. Thuy and D. V. Sang, "NeoUNet : Towards Accurate Colon Polyp Segmentation and Neoplasm Detection," in *Advances in Visual Computing, ISVC 2021*, 2022, pp. 15-28.

[13] D. Jha, P. H. Smedsrud, M. A. Riegler, P. Halvorsen, T. de Lange, D. Johansen and H. D. Johansen, "Kvasir-SEG: A Segmented Polyp Dataset," *MultiMedia Modeling,* pp. 451-462, 2019.

[14] J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, D. Gil, C. Rodríguez and F. Vilariño, "WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians," *Computerized medical imaging and graphics,* pp. 99-111, 2015.

[15] N. Tajbakhsh, S. R. Gurudu and J. Liang, "Automated Polyp Detection in Colonoscopy Videos Using Shape and Context Information," *IEEE transactions on medical imaging,* vol. 35, no. 2, pp. 630-644, 2016.

[16] J. Silva, A. Histace, O. Romain, X. Dray and B. Granado, "Toward embedded detection of polyps in WCE images for early diagnosis of colorectal cancer," *International journal for computer assisted radiology and surgery,* vol. 9, no. 2, pp. 283-293, 2013.

[17] D. Vázquez, J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, A. M. López, A. Romero, M. Drozdzal, A. Courville and J. Gao, "A Benchmark for Endoluminal Scene Segmentation of Colonoscopy Images," *Journal of healthcare engineering,* pp. 4037190-9, 2017.

[18] A. Vannelli, *Colorectal Cancer*, London, UK: IntechOpen, 2021.

[19] D. Swinson, M. Seymour and R. A. Adairm, *Colorectal cancer,* Oxford: Oxford University Press, 2012.

[20] O. Engin, *Colon Polyps and the Prevention of Colorectal Cancer,* Buca, Turkey: Springer, 2015.

[21] M.S. Cappell, "Reducing the Incidence and Mortality of Colon Cancer: Mass Screening and Colonoscopic Polypectomy," *Gastroenterology clinics of North America,* vol. 37, no. 1, pp. 129-160, 2008.

[22] D. K. Rex, "Reducing costs of colon polyp management," *The Lancet Oncology,* vol. 10, no. 12, pp. 1135-1136, 2009.

[23] A. G. Zauber, S. J. Winawer, M. J. O'Brien, I. Lansdorp-Vogelaar, M. van Ballegooijen, B. F. Hankey, W. Shi, J. H. Bond, M. Schapiro, J. F. Panish, E. T. Stewart and J. D. Waye, "Colonoscopic polypectomy and long-term prevention of colorectal-cancer deaths," *New England journal of medicine,*, vol. 366, no. 8, pp. 687-696, 2012.

[24] M. Morris, B. Iacopetta and C. Platell, "Comparing survival outcomes for patients with colorectal cancer treated in public and private hospitals," *Medical Journal of Australia*, vol. 186, no. 6, pp. 296-300, 2007.

[25] R. L. Siegel, K. D. Miller, S. A. Fedewa, D. J. Ahnen, R. G. S. Meester, A. Barzi and A. Jemal, "Colorectal cancer statistics, 2017," *CA: a cancer journal for clinicians,* vol. 67, no. 3, pp. 177-193, 2017.

[26] T. Matsuda, A. Ono, Y. Kakugawa, M. Matsumoto and Y. Saito, "Impact of screening colonoscopy on outcomes in colorectal cancer," *Japanese Journal of Clinical Oncology*, vol. 45, no. 10, pp. 900-905, 2015.

[27] Y. Cha, W. Choi and O. Büyüköztürk, "Deep Learning-Based Crack Damage Detection Using Convultional Neural Networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 361-378, p. 21, 2017.

[28] A. Titoriya and S. Sachdeva, "Breast Cancer Histopathology Image Classification using AlexNet," in *2019 4th International Conference on Information Systems and Computer Networks (ISCON), November, 2019*, Mathura, India [Online]. Available: IEEE Xplore, https://ieeexplore.ieee.org/abstract/document/9036160.

[29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going Deeper with Convolutions," 2014, *arXiv:1409.4842*. [Online]. Available: https://arxiv.org/abs/1409.4842

[30] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image," 2014, *arXiv:1409.1556.* [Online]. Available: https://arxiv.org/abs/1409.1556

[31] E. Shelhamer, J. Long and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," 2015, *arXiv:1411.4038*. [Online]. Available: https://arxiv.org/abs/1411.4038

[32] O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015,* 2015. [Online]. Available: Springer, https://link-springer-com.uml.idm.oclc.org/chapter/10.1007/978-3-319-24574-4_28

[33] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," in *Proceedings of the European conference on computer vision (EECV),* Springer International Publishing, 2018. pp. 833-851.

[34] D. Jha, S. Ali, N. K. Tomar, H. D. Johansen, D. Johansen, J. Rittscher, M. A. Riegler and P. Halvorsen, "Real-Time Polyp Detection, Localization and Segmentation in Colonoscopy Using Deep Learning," *IEEE Access*, vol. 9, pp. 40496-40510, 2021.

[35] L.-C. Chen, G. Papandreou, F. Schroff and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: https://arxiv.org/abs/1706.05587

[36] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016*, [Online]. Available: IEEE Xplore, https://ieeexplore.ieee.org/document/7780719

[37] Y. Komeda, H. Handa, T. Watanabe, T. Nomura, M. Kitahashi, T. Sakurai, A. Okamoto, T. Minami, M. Kono, T. Arizumi, M. Takenaka, S. Hagiwara, S. Matsui and N. Nishida, "Computer-Aided Diagnosis Based on Convolutional Neural Network System for Colorectal Polyp Classification: Preliminary Experience," *Oncology,* vol. 93, no. 1, pp. 30-34, 2017.

[38] M. L. a. N. K. Sungheon Park, "Polyp detection in Colonoscopy Videos Using Deeply-Learned Hierarchical Features," in *ISBI 2015 Grand Challenge on Automatic Polyp Detection in Colonoscopy videos, 2015, Seoul.* [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.718.7955&rep=rep1&type=pdf.

[39] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba and A. Oliva, "Learning Deep Features for Scene Recognition using Places Database," in *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, 2014. pp. 487-495.

[40] J. Y. Lee, J. Jeong, E. M. Song, C. Ha, H. J. Lee, J. E. Koo, D.-H. Yang, N. Kim and J.-S. Byeon, "Real-time detection of colon polyps during colonoscopy using deep learning: systematic validation with four independent datasets," *Scientific Reports*, vol. 10, no. 1, pp. 8379-8379, 2020.

[41] L. Yu, H. Chen, Q. Dou, J. Qin and P. A. Heng, "Integrating Online and Offline Three-Dimensional Deep Learning for Automated Polyp Detection in Colonoscopy Videos," *IEEE journal of biomedical and health informatics*, vol. 21, no. 1, pp. 65-75, 2016.

[42] A. Sánchez-González, B. García-Zapirain, D. Sierra-Sosa and A. Elmaghraby, "Automatized colon polyp segmentation via contour region analysis," *Computers in biology and medicine*, vol. 100, pp. 152-164, 2018.

[43] J. Kang and J. Gwak, "Ensemble of Instance Segmentation Models for Polyp Segmentation in Colonoscopy Images," *IEEE*, vol. 7, pp. 26440-26447, 2019.

[44] X. Guo, N. Zhang, J. Guo, H. Zhang, Y. Hao and J. Hang, "Automated polyp segmentation for colonoscopy images: A method based on convolutional neural networks and ensemble learning," *Medical physics*, vol. 46, no. 12, pp. 5666-5676, 2019.

[45] Y. Zhang, H. Liu and Q. Hu, "TransFuse: Fusing Transformers and CNNs for Medical Image Segmentation," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*, 2021, pp. 14-24.

[46] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Computer Vision – ECCV 2014*, 2014. p.740-755.

[47] V. Badrinarayanan, A. Kendall and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481-2495, 2017.

[48] H. Zhao, J. Shi, X. Qi, X. Wang and J. Jia, "Pyramid Scene Parsing Network,", 2016, *arXiv:1612.01105*. [Online]. Available: https://arxiv.org/abs/1612.01105.

[49] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles and H. Jégou, "Training data-efficient image transformers & distillation through attention,", 2020, *arXiv:2012.12877*. [Online]. Available: https://arxiv.org/abs/2012.12877.

[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770-778.

[51] S. Ali, F. Zhou, B. Braden, A. Bailey, S. Yang, G. Cheng, P. Zhang, X. Li, M. Kayser, R. D. Soberanis-Mukul, S. Albarqouni, X. Wang, C. Wang, S. Watanabe and I. Oksuz, "An objective comparison of detection and segmentation algorithms for artefacts in clinical endoscopy," *Scientific Reports*, vol. 10, no. 1, pp. 1-15, 2020.

[52] T. Mahmud, B. Paul and S. A. Fattah, "PolypSegNet: A modified encoder-decoder architecture for automated polyp segmentation from colonoscopy images," *Computers in biology and medicine*, vol. 128, pp. 104119-, 2021.

[53] J. Hu, L. Shen, S. Albanie, G. Sun and E. Wu, "Squeeze-and-Excitation Networks," *IEEE transactions on pattern analysis and machine intelligence,* pp. 2011 - 2023, 2020.

[54] A. A. Pozdeev, N. A. Obukhova and A. A. Motyko, "Automatic analysis of endoscopic images for polyps detection and segmentation," in *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Saint Petersburg and Moscow, Russia,* 2019, pp. 1216–1220.

[55] D.-P. Fan, G.-P. Ji, T. Zhou, G. Chen, H. Fu, J. Shen and L. Shao, "PraNet: Parallel Reverse Attention Network for Polyp Segmentation," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020, Lima, Peru,* 2020, pp. 263–273.

[56] S. Chen, X. Tan, B. Wang and X. Hu, "Reverse Attention for Salient Object Detection," in *Computer Vision – ECCV 2018,* 2018, pp. 236-252.

[57] M. Cheng, Z. Kong, G. Song, Y. Tian, Y. Liang and J. Chen, "Learnable Oriented-Derivative Network for Polyp Segmentation," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021,* 2021, p.720-730.

[58] P. Song, J. Li and H. Fan, "Attention based multi-scale parallel network for polyp segmentation," *Computers in Biology and Medicine,* vol. 146, pp. 105476-, 2022.

[59] B. Kayalibay, G. Jensen and P. van der Smagt, "CNN-based Segmentation of Medical Imaging Data,", 2017. *arXiv:1701.03056v2.* [Online]. Available: https://arxiv.org/pdf/1701.03056.pdf

[60] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh and J. Liang, "UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation," *IEEE transactions on medical imaging*, vol. 39, no. 6, pp. 1856-1867, 2019.

[61] Y. Wang, N. Wang, M. Xu, J. Yu, C. Qin, X. Luo, X. Yang, T. Wang, A. Li and D. Ni, "Deeply-Supervised Networks With Threshold Loss for Cancer Detection in Automated Breast Ultrasound," *IEEE transactions on medical imaging*, pp. 866-876, 2020.

[62] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit and N. Houlsby, "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale,", 2020 *arXiv:2010.11929* , [Online]. Available: https://arxiv.org/abs/*2010.11929*

[63] F. Chollet, "Xception: Deep learning with depthwise separable convolutions,", 2017, *arXiv:1610.02357.* [Online]. Available: https://arxiv.org/abs/1610.02357

[64] D. Kang and Y. J. Cha, "Efficient attention-based deep encoder and decoder for automatic crack segmentation," *Structural health monitoring*, 2021.

[65] R. Ali and Y. J. Cha, "Attention-based generative adversarial network with internal damage segmentation," *Automation in Construction*, vol. 141, 2022.

[66] D. Hendrycks and K. Gimpel, "Gaussian Error Linear Units (GELUs),", 2016, *arXiv:1606.08415.* [Online]. Available: https://arxiv.org/abs/1606.08415

[67] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, "Attention Is All You Need,", 2017, *arXiv:1706.03762*. [Online]. Available: https://arxiv.org/abs/1706.03762

[68] R. Strudel, R. Garcia, I. Laptev and C. Schmid, "Segmenter: Transformer for Semantic Segmentation," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. [Online]. Available: IEEE Xplore, https://ieeexplore-ieee-org.uml.idm.oclc.org/document/9710959

[69] P. Jaccard, "The Distribution of the Flora in the Alpine Zone," *The New phytologist*, vol. 11, no. 2, pp. 37-50, 1912.

[70] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid and S. Savarese, "Generalized Intersection over Union,", 2019, *arXiv:1902.09630*. [Online]. Available: https://arxiv.org/abs/1902.09630

[71] L. R. Dice, "Measures of the Amount of Ecologic Association Between Species," *Ecology*, pp. 297-302, 1945.