University of Nottingham
UK | CHINA | MALAYSIA

PHD THESIS

Towards Real-World Clinical Colonoscopy deep learning Models for Video-based Bowel Preparation and Generalisable Polyp Segmentation

Student:
Mahmood Haithami

Supervisor:
Dr Iman Yi Liao

Co-Supervisors:
Dr Amr Ahmed
Dr Hafeez Ullah Amin

Faculty of Science and Engineering

School of Computer Science

March 2024

Acknowledgment

First and foremost, I want to thank Allah Almighty for what he has given me without any power or strength on my part. He gave me sound thinking and then facilitated for me the sources of success. Without him life has no meaning to me! Praise be to Allah first and last.

My special thanks go to my dear father, who raised me and supported me financially and morally. The amount he spent on me from my birth until today is a fortune, but he chose to invest in me. Literally, my father did not neglect anything at all, but rather removed the financial obstacles from me completely. I never felt nervous or anxious under him. I hope his investment succeeds one day. I also want to thank the one who was patient with me in her womb for nine months of pregnancy and then nurtured me until my youth. My mom. She was the one who taught me when I was a kid, and truth be told, I made her tired during high school because I did not care about my studies. At a moment, she thought that academic studies were not suitable for me. She was worried about my future and thought that my future would have nothing to do with studying/academia! She didn't share with me what was on her mind at the time, but I found out nearly a decade later by coincidence. Today I dedicate this achievement to her. I also want to thank my wife and daughter for their patience with me throughout this period. They were a source of comfort for me and positive distractions.

Fortunately for me, I was blessed with more than wonderful academic supervisors, Dr Iman and Dr Amr. During my PhD, I saw nothing but unlimited support and attention from them. I did not hesitate to contact them at any time. They allocated more than an hour of their time to me every week throughout my doctoral studies. I can't remember the times a meeting was cancelled because they were busy! Though they are actually very busy! The opposite was true, as I was the one who often cancelled the meetings. I don't know where they get this patience and persistence. Their criticisms were soft and indirect, and I never heard negative words from them. I think I have good luck!

I want to thank those who spent quite a bit of time reading my thesis, internal examiner Dr Tissa and the external examiner Dr Nasharuddin. They could have had an enjoyable time doing anything other than reading 150 technical pages! I thank them for their patience and comments that contributed toward improving my thesis. I cannot forget Dr Tomas for being the chair during my Viva. He managed the Viva session professionally. He also one of the first staff I met in UNM. He has pleasant and positive character. Also, all gratitude to the administrative staff, including Asyiqin and Sharon, for their encouragement and providing moral support, especially on the day of the thesis defence. I also want to thank the undergraduate students whom I taught during my doctoral studies. Teaching sessions relieved my stress and made me forget some of the obstacles I faced during my PhD. To be in the safe side, I would like to thank all the people who gave me any kind of positivity during my PhD. Thank you all!

Finally, I want to thank you, the University of Nottingham, Malaysia, for giving me the opportunity to obtain a doctoral scholarship and providing all the important training courses in academic and practical life. I hope that the university will keep flourishing and provide opportunities for future students to do their research and solve real world problems.

Publications

- M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab, "**An embedded recurrent neural network-based model for endoscopic semantic segmentation.**" *In Proceedings of the 3rd International Workshop and Challenge on Computer Vision in Endoscopy (EndoCV 2021) co-located with the 18th IEEE International Symposium on Biomedical Imaging (ISBI 2021), Nice, France, April 13, 2021, vol. 2886, 49–58 (CEUR-WS.org, 2021).*

- M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab, "**Employing GRU to combine feature maps in DeeplabV3 for a better segmentation model.**" *Nordic Machine Intelligence, vol. 1, no. 1, pp. 29–31, Nov. 2021.*

- Ali, Sharib, et al., "**Assessing generalisability of deep learning-based polyp detection and segmentation methods through a computer vision challenge.**" *Sci Rep 14, 2032 (2024).*

- M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab, "**Automatic bowel preparation assessment using deep learning.**" *In: Rousseau, JJ., Kapralos, B. (eds) Pattern Recognition, Computer Vision, and Image Processing. ICPR 2022 International Workshops and Challenges. ICPR 2022. Lecture Notes in Computer Science, vol 13643. Springer, Cham.*

- Eisenmann, Matthias, et al. "**Biomedical image analysis competitions: The state of current participation practice.**" *Preprint at arXiv:2212.08568 (2022).*

- M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab. "**Enhancing Polyp Segmentation Generalisability by Minimizing Images' Total Variation.**" *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2023.

- **(Accepted-best paper award)** M. Haithami, A. Ahmed, I. Y. Liao**. "Enhancing generalisability of deep learning polyp segmentation using online spatial interpolation and hue transformation.**" *Brain Inspired Cognitive Systems: 11th International Conference, BICS 2023, Kuala Lumpur, Malaysia, 2023, Proceedings 11. Lecture Notes in Computer Science, Springer.*

- **(Under Review)** M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab. "**Bowel preparation assessment using deep learning: a solved problem or yet to be solved?**" *IEEE Transactions on Medical Imaging.*

Abstract

Colorectal cancer is the most prevalence type of cancers within the digestive system. Early screening and removal of precancerous growths in the colon decrease mortality rate. The golden standard screening type for colon is colonoscopy which is conducted by a medical expert (i.e., colonoscopist). Nevertheless, due to human biases, fatigue, and experience level of the colonoscopist, colorectal cancer missing rate is negatively affected. Artificial intelligence (AI) methods hold immense promise not just in automating colonoscopy tasks but also enhancing the performance of colonoscopy screening in general. The recent development of intense computational GPUs enabled a computational-demanding AI method (i.e., deep learning) to be utilised in various medical applications. However, given the gap between the clinical-practice and the proposed deep learning models in the literature, the actual effectiveness of such methods is questionable. Hence, this thesis highlights such gaps that arises from the separation between the theoretical and practical aspect of deep learning methods applied to colonoscopy. The aim is to evaluate the current state of deep learning models applied in colonoscopy from a clinical angle, and accordingly propose better evaluation strategies and deep learning models. The aim is translated into three distinct objectives. The first objective is to develop a systematic evaluation method to assess deep learning models from a clinical perspective. The second objective is to develop a novel deep learning architecture that leverages spatial information within colonoscopy videos to enhance the effectiveness of deep learning models on real-clinical environments. The third objective is to enhance the generalisability of deep learning models on unseen test images by developing a novel deep learning framework. To translate these objectives into practice, two critical colonoscopy tasks, namely, automatic bowel preparation and polyp segmentation are attacked. In both tasks, subtle overestimations are found in the literature and discussed in the thesis theoretically and demonstrated empirically. These overestimations are induced by improper validation sets that would not appear or represent the real-world clinical environment. Arbitrary dividing colonoscopy datasets to do deep learning evaluation can result in producing similar distributions, hence, achieving unrealistic results. Accordingly, these

factors are considered in the thesis to avoid such subtle overestimation. For the automatic bowel preparation task, colonoscopy videos that closely resemble clinical settings are considered as input and accordingly it necessitates the design of the proposed model as well as evaluation experiments. The proposed model's architecture is designed to utilise both temporal and spatial information within colonoscopy videos using Gated Recurrent Unit (GRU) and a proposed Multiplexer unit, respectively. Meanwhile for the polyp segmentation task, the efficiency of current deep learning models is tested in terms of their generalisation capabilities using unseen test sets from different medical centres. The proposed framework consists of two connected models. The first model is responsible for gradually transforming textures of input images and arbitrary change their colours. Meanwhile the second model is a segmentation model that outlines polyp regions. Exposing the segmentation model to such transformed images acquires the segmentation model texture/colour invariant properties, hence, enhances the generalisability of the segmentation model. In this thesis, rigorous experiments are conducted to evaluate the proposed models against the state-of-the-art models. The yielded results indicate that the proposed models outperformed the state-of-the-art models under different settings.

Contents

List of figures

List of tables

Acronyms

A	**ACG**	: American College of Gastroenterology
	ADR	: Adenoma detection rate
	AI	: Artificial Intelligence
	ASM	: Adaptive Selection Module
	ASGE	: Gastrointestinal Endoscopy
	Avg	: Average
B	**BBPS**	: Boston Bowel Preparation Scale
C	**CAD**	: Computer-aided system
	CADe	: Computer-aided detection
	CADm	: Computer-aided monitoring
	CADx	: Computer-aided diagnosis
	CaraNet	: Context Axial Reverse Attention Network
	CNN	: Convolutional Neural Network
D	**DL**	: deep learning
	EAM-Net	: Efficient Attention Mechanism Network
E	**EGD**	: Upper Endoscopy/Esophagogastroduodenoscopy
F	**FC**	: Fully Connected layer
	FCN	: Fully Convolutional Network
	FN	: False Negative
	FP	: False Positive
	FS-CRF	: Frame Search Conditional Random Field
G	**GANs**	: Generative Adversarial Networks
	GCM	: Global Context Module
	GF	: Global Features
	GI	: Gastrointestinal tract
	GRU	: Gated Recurrent Unit
I	**IoU**	: Intersection over Union
	mIoU	: mean IoU
L	**LCA**	: Local Context Attention
	LSTM	: Long Short-Term Memory
M	**ML**	: Machine Learning
	MLP	: Multilayer Perceptron
N	**NLP**	: Natural Language Processing
P	**PCA**	: Principle Component Analysis
R	**ReLU**	: Rectified Linear Unit
	RGB	: Red-Green-Blue
	RNN	: Recurrent Nural Network
S	**SOTA**	: State-Of-The-Art
	SVM	: Support Vector Machine
T	**t-SNE**	: t-distributed Stochastic Neighbour Embedding
	TI	: Texture Interpolation
	TL	: Transfer Learning
	TN	: True Negative
	TP	: True Positive
	TV	: Total Variation
V	**ViT**	: Vision Transformer

Chapter 1

Introduction

The gastrointestinal tract, depicted in Figure 1, forms the pathway of the digestive system, extending from the mouth to the anus. It comprises vital digestive organs such as the oesophagus, stomach, small intestines, and large intestines (colon).

As per the National Institute of Diabetes and Digestive and Kidney Diseases [1], around 60 to 70 million individuals suffer from gastrointestinal diseases. Additionally, the Gastrointestinal (GI) tract is associated with eight of the most prevalent cancers [2]. Notably, in 2020, colon cancer alone was identified as the second leading cause of cancer-related deaths worldwide by the World Health Organization [3]. Consequently, regular screening of the gastrointestinal tract, with a specific focus on the colon, will lead to a reduction in mortality rates.

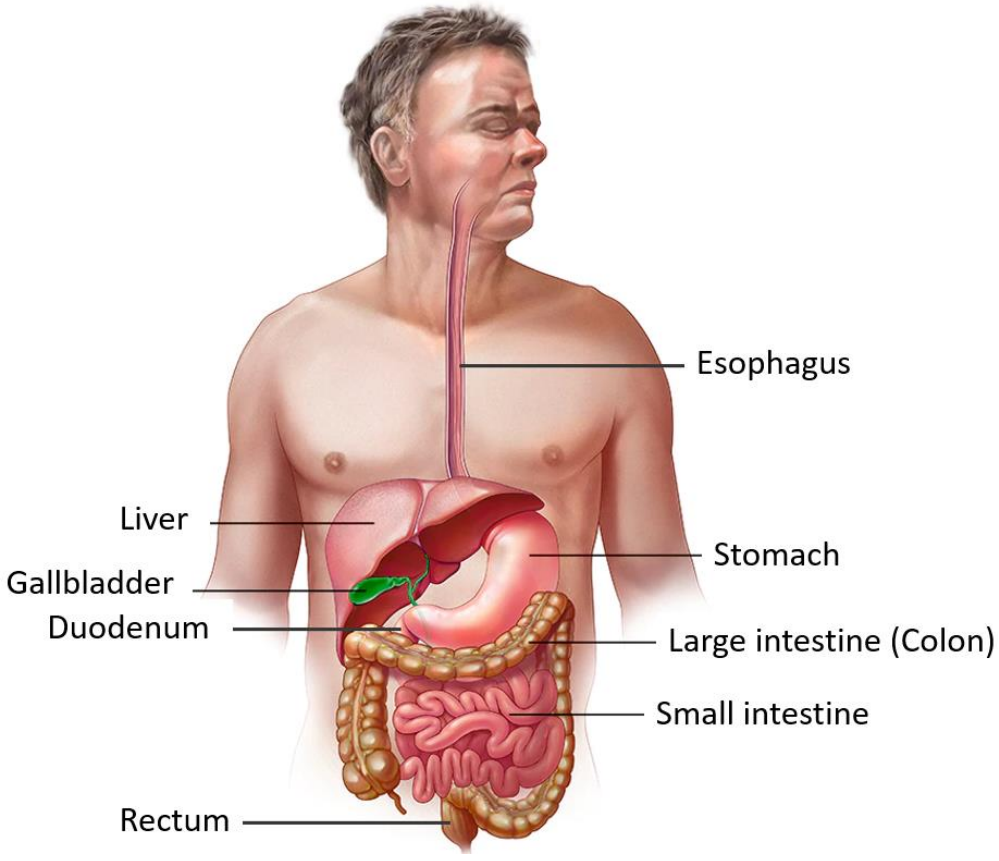


Figure 1 Gastrointestinal tract anatomy. Original image was taken from [4].

The primary approach for investigating the gastrointestinal (GI) tract is through endoscopy [2]. However, endoscopists are facing challenges due to the substantial workload involved in analysing various types of endoscopic images [5]. To facilitate timely GI treatment and prevent further complications, it

becomes essential to detect diseases in the inner lining of the gastrointestinal tract (i.e., mucosa). Therefore, the implementation of a Computer-aided detection (CAD) system holds significant value for assisting endoscopists in assessing severity and identifying abnormalities [2]. More discussion on CAD can be found in section 1.1.2.

Thanks to modern-day computers, CAD systems demonstrate highly enhanced pattern-detection capabilities when processing databases [6]. Artificial Intelligence (AI) methods employed in this field can be broadly categorized into conventional Machine Learning (ML) approaches (such as K-NN and SVM) and deep learning (DL) methods such as ResNet [7], U-Net [8], and Faster-R-CNN [9].

Recently, DL methods have garnered significant attention from researchers due to their ability to outperform hand-crafted classifiers (i.e., ML approaches) [10], [11]. However, DL methods necessitate a substantial amount of training datasets to achieve exceptional performance. Regrettably, in the domain of Gastrointestinal endoscopy, as well as in the medical domain in general, there is a scarcity of such datasets. The lack of extensive and publicly available datasets is attributed to the high cost of manual annotation and privacy concerns[12], [13]. The absence of an adequate database poses a real challenge in developing efficient systems based on DL.

To address this limitation, techniques such as image augmentation, Generative Adversarial Networks (GANs), Zero/One-shot learning, and Transfer Learning have been proposed in the literature to enhance the overall performance of DL methods and mitigate the impact of having a small training dataset [14]–[16].

Nevertheless, creating an effective DL model for the Gastrointestinal (GI) domain remains a challenging task due to the unique nature of the tissues, lacking specific shapes or clear edges. Additionally, the high interclass similarity further complicates the development of such models.

1.1 Technical and medical background

This section provides a presentation of both medical and technical backgrounds. Additionally, it will explain terminologies and specialised language used in the field of the Gastrointestinal tract (GI) to aid discussions in subsequent sections. Initially, an introduction to various types of GI endoscopy, accompanied by informative visuals, will be presented. Following that, different types of computer-aided (CAD) systems mentioned in existing literature will be discussed. Lastly, an overview of the applications of deep learning in the GI domain will be summarized and demonstrated.

1.1.1 Gastrointestinal tract anatomy and types of endoscopic screening

Gastrointestinal tract (GI) or the digestive system is the tract/pathway by which the food goes through starting from the mouth followed by oesophagus, stomach, small intestine, colon and finally the rectum in which the waste/stool is discarded, as depicted in Figure 1.

Endoscopy is a medical procedure that utilises an instrument called an endoscope to examine the internal organs or pathways within the body [17]. An endoscope is a tube-like device equipped with a lens and a light source for visualisation. However, the term "endoscopy" is sometimes used interchangeably with GI endoscopy, although there is no universal consensus on its precise usage [18]. It should be noted that endoscopy is not limited to the GI tract and can also be employed in various other areas of the body, such as the nose, ears, heart, and joints. For more comprehensive information regarding the different types of endoscopy and the technologies available, consult the following papers [19]–[21].

The classification of endoscopy types is determined by the specific procedure being performed, the targeted area or organ, and the approach used to reach the desired location. For instance, Bronchoscopy is utilised to examine the lungs and involves inserting the endoscope through the mouth, while Arthroscopy is employed for joint surgeries and involves inserting the endoscope through a small incision near the joint. In the realm of gastrointestinal screening, the following are among the most commonly used types of endoscopy, as referenced in [18], [22], [23]:

- **Esophagoscopy**: An esophagoscope is inserted through the mouth to examine the oesophagus.
- **Gastroscopy**: A gastroscope is inserted through the mouth to examine stomach and duodenum. Duodenum is the beginning of the small intestine.
- **Colonoscopy**: A colonoscope is inserted through the anus to examine the entire length of the colon and large intestine. Colonoscopy is the golden standard for colon screening, as seen in Figure 2. Colonoscopy is the main research area targeted in this thesis.
- **Proctoscopy:** A proctoscope is inserted through the anus to examine the bottom part of the colon which consists of rectum and sigmoid colon.
- **Sigmoidoscopy**: A sigmoidoscope is inserted through the anus to examine only the sigmoid colon.
- **Wireless Capsule Endoscopy:** • Wireless Capsule Endoscopy is a compact device that captures video footage and transmits it to a receiver worn on the patient's waist. This capsule is designed for single-use and is primarily utilised as a diagnostic tool. Its main application lies in

diagnosing conditions affecting the small intestine, which can be challenging or even inaccessible using conventional endoscopic procedures.

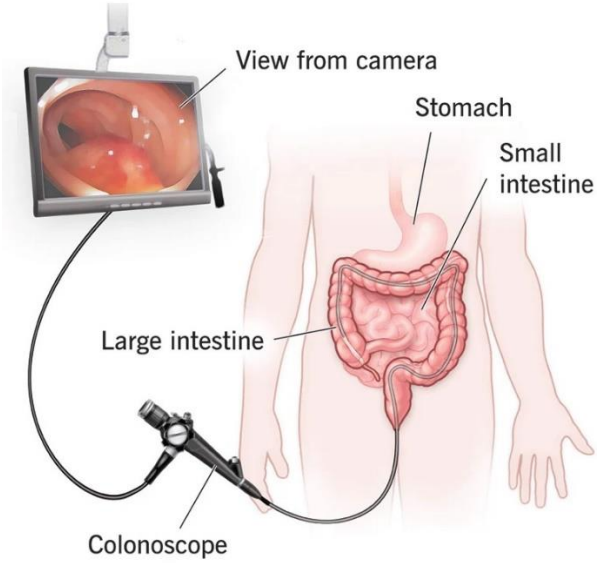


Figure 2 Colonoscopy screening. Image courtesy of [24].

A laparoscope is an endoscope specifically designed to be inserted through a small surgical incision in the abdomen. The procedure performed using a laparoscope is called laparoscopy [18]. Laparoscopy can be employed to examine and operate on various organs within the abdominal region, including the stomach, liver, and other abdominal organs. This minimally invasive surgical technique is often referred to as "minimally-invasive surgery". The field of robotics and computer vision has shown particular interest in laparoscopy due to its potential for three-dimensional organ reconstruction, instrument detection, and tracking. These advancements aim to enable robot-assisted surgery [18].

To provide a comprehensive overview of GI endoscopy procedures and their interrelationships, Figure 3 has been created. This figure visually depicts the target areas for screening within the GI domain, along with the corresponding procedure names. GI endoscopy can be broadly classified into two main types:

- Upper Endoscopy or Esophagogastroduodenoscopy (EGD): This procedure involves the examination of the oesophagus, stomach, and duodenum.
- Colonoscopy (also informally known as lower endoscopy): This procedure allows for the thorough examination of the entire colon.

It's important to note that conventional endoscopy can only reach the initial part of the small intestine, known as the duodenum. In order to examine the rest of the small intestine, a specialised device called Enteroscopy is employed. Enteroscopy utilises a distinct type of endoscope that employs various mechanisms to control its movement within the GI tract. Examples of such

mechanisms include balloon-assisted Enteroscopy and spiral overtube-assisted Enteroscopy [23], [25].

However, due to its limitations and potential complications, Wireless Capsule Endoscopy (WCE) is the commonly used method for investigating the small intestine. WCE involves using a capsule-shaped endoscope that can capture images as it traverses through the digestive system. It's important to note that WCE is primarily considered a diagnostic tool, whereas Enteroscopy is recognized as both a diagnostic and therapeutic device [23].

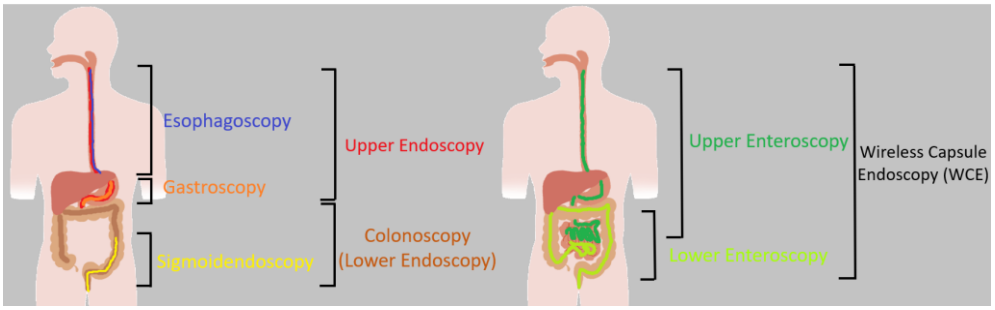


Figure 3 illustrates the various screening types (i.e., endoscopy) categorized according to the specific targeted areas within the gastrointestinal tract (GI).

Based on the provided information, this study is centered around the implementation of deep learning methods specifically for colonoscopy. Consequently, other procedures such as Laparoscopy, Cystoscopy (used for bladder examinations), Arthroscopy and Gastroscopy are beyond the scope of this research. The primary emphasis is on exploring the utilisation of deep learning techniques in the context of colonoscopy screenings.

1.1.2 Types of CAD systems

CADe, CADx, and CADm are abbreviations used to distinguish the purpose or type of Computer-Aided Diagnosis (CAD) systems. However, there can be confusion and overlap between these terms. Generally, the primary objective of these systems is to enhance the performance of an endoscopist during or after a screening procedure. To clarify the differences between them, explanations are listed as follows [26]:

- CADe: Computer-aided detection is a CAD system designed to assist in the identification of abnormal regions within medical images or videos. It focuses on detecting potential abnormalities but leaves the determination of severity or grading to the endoscopist.
- CADx: Computer-aided diagnosis aims to support endoscopists in making optical diagnoses and determining the severity or type of detected abnormalities. It goes beyond detection and provides additional information to aid in the diagnostic process and targeted biopsies. Combining CADe and CADx systems can help endoscopists effectively discriminate, diagnose, and disregard abnormalities [27].
- CADm: Computer-aided monitoring system is used to evaluate and monitor the performance of procedures or endoscopists. Its tasks include ensuring the completeness of inspections, supervising examination time,

and enhancing mucosal visibility. Additionally, in the Multimedia community, CADm systems can provide summaries for recorded endoscopic videos, which is of particular interest for reviewing and analysing procedures [18].

In the literature, the term "CAD" (Computer-Aided Diagnosis) is often used in a broad sense to indicate that a computer system is employed or proposed to aid or evaluate physicians, regardless of the specific medical procedures involved. CAD systems can be versatile and applied in various medical fields (e.g., clinical pathology, radiology, ophthalmology, and dermatology) to augment healthcare professionals' capabilities [28].

It is essential to note that CAD systems can be developed using different methodologies, such as Artificial Intelligence (AI) methods or Image processing techniques. While both approaches have their merits, AI-based CAD systems are particularly powerful and have the capacity to tackle more complex problems. Leveraging AI (i.e., deep learning), these systems can learn from large datasets, recognize patterns, and provide more advanced decision support to healthcare practitioners, thus improving overall diagnostic accuracy and patient care. Nonetheless, traditional image processing methods still play a valuable role in some applications where AI might be less suitable or necessary.

1.1.3 Application of deep learning in GI endoscopy screening

There are three main applications for deep learning (D) in the Gastrointestinal tract (GI) analysis tasks [29], namely, classification, detection, and segmentation as seen in Figure 4. These categories are explained as follows:

a. **Classification**: In this task, an input image is assigned a specific label. For instance, the image may be categorized as either "normal" or "abnormal." However, unlike detection and segmentation, the precise location of the abnormality within the image is not provided.
b. **Detection**: In this task, abnormalities within an image are identified and framed using bounding-boxes. These bounding-boxes overlay the image, capturing the attention of endoscopists to further investigate the identified locations. The input is an image, and the output comprises the centre point and dimensions of the bounding-box (e.g., Height and Width).
c. **Segmentation**: This process can be regarded as a detailed form of detection, where each pixel within the image is assigned a specific label. The label "0" represents normal regions, while label "1" designates abnormal areas. Unlike detection, segmentation provides a precise outline of the abnormal regions in the image, enabling more accurate localisation of affected areas.

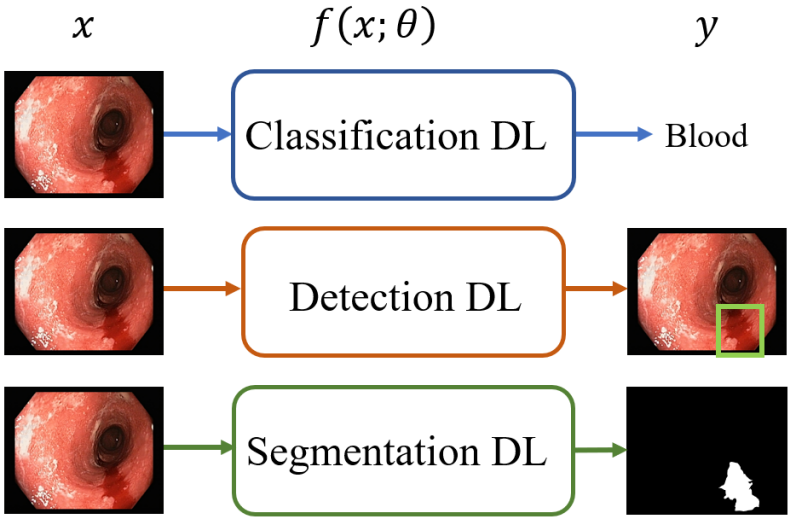


Figure 4 illustrates the application of a CAD system for GI screening and diagnosis [29].

Classification tasks can be viewed as image-level labelling, while segmentation tasks can be seen as pixel-level classification. On the other hand, detection tasks are akin to regression problems, where the output consists of real numbers representing the box borders that identify the detected abnormalities.

1.2 Challenges of deep learning applied in colonoscopy

Observations from the existing literature reveal that a significant portion of the research conducted in the field of colonoscopy has predominantly utilised carefully selected high-quality still images, as opposed to video data. This choice of using carefully selected images contrasts with the reality of colonoscopy, where challenges such as image blurring and poor quality are commonplace. Moreover, the real-time scenario involves a multitude of ambiguous and non-informative frames that need to be considered during the design of a deep learning (DL) model [30]. Failing to account for temporal frames leads to the loss of a crucial form of contextual awareness, which otherwise contributes to improved decision-making capabilities [30].

Moreover, a notable absence of standardized benchmarks that could facilitate performance comparisons is evident. Consequently, a significant portion of researchers tend to construct their training and validation datasets in an arbitrary manner, thereby hindering the possibility of cross-performance evaluations and hampering the reproducibility of research outcomes. What's more, this ad-hoc dataset preparation method could potentially result in a subtle inflated performance. To elaborate, while a system might yield impressive outcomes during validation, its performance could drastically decline when tested on unseen datasets or when put into practical use (referred to as the "generalisability problem" [31].

In response to this issue, a cohort of researchers initiated a colonoscopy challenge that is explicitly tailored to evaluate generalisability. This challenge provides training and validation sets while withholding a separate test set [32]. Consequently, participants are required to submit their trained models (checkpoints), which are subsequently assessed by the challenge organizers. Such challenges serve as effective mechanisms for assessing the generalisability capabilities of deep learning models. However, it is worth noting that these challenges often exhibit limited scale, both in terms of participants and the datasets used, especially when compared to other domains outside of the medical field.

Furthermore, ensuring the reproducibility and transparency of deep learning models within the medical domain remains uncertain, even when the source code is made available [33], [34]. Addressing this concern, the Norwegian Artificial Intelligence Research Consortium (NORA) orchestrated a challenge aimed at evaluating not only the performance of deep learning models but also the transparency procedures implemented or suggested by participants [35]. Initiatives of this nature, which strive for both reproducibility and transparency, are relatively uncommon in the medical domain as a whole, and particularly within the field of colonoscopy.

Consequently, contemporary journals and conference proceedings have started to encourage researchers to share their source codes as a means to facilitate research reproducibility. Nonetheless, certain published studies are based on proprietary datasets [36]–[39], rendering reproducibility unattainable. For privacy considerations, researchers are often unable to disclose these private datasets. Furthermore, even if the datasets used are publicly accessible, achieving reproducibility is not guaranteed due to the vague definition and arbitrary creation of training and validation sets.

Considering the remarkable achievements of deep learning in domains like natural images (e.g., ImageNet), it seems reasonable to assume that such success might extend to the field of colonoscopy. However, disregarding the distinctive characteristics of colon mucosa (i.e., the inner surface of the colon), which markedly differs from non-medical domain images, could result in subpar performance. Unlike the clear and well-defined shapes commonly found in non-medical domains, colonoscopy images lack such structure (i.e., for instance look at Figure 5). Natural images have well defined borders and shape compared to colon mucosa. Within the same class, colonoscopy image could look very different and vice versa. This presents a formidable challenge that necessitates researchers to incorporate these unique properties into the design of their deep learning models.

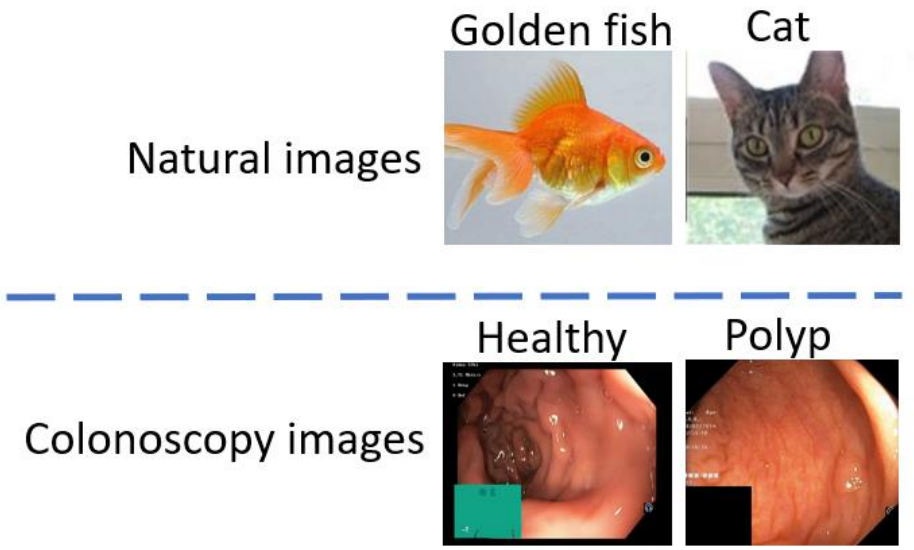


Figure 5 Samples from natural images compared against colonoscopy.

In the broader context, the prevailing trend in deep learning for non-medical domains involves constructing increasingly complex architectures to accommodate vast datasets. Applying these deep learning models as-is, rather than developing specialised architectures, might lead to overfitting due to the relatively limited size of available medical datasets compared to their non-medical counterparts.

1.3 Thesis scope, aim, and objectives

The digestive system encompasses various organs, including the throat, oesophagus, stomach, small intestine, large intestine (colon), and rectum, as depicted in Figure 1. Within this system, colonoscopy is recognized as the primary screening method for the colon; refer to the preceding section for visual representations of these terms. Consequently, the central focus of this thesis revolves around the application of deep learning techniques in the context of colonoscopy, as seen in Figure 6.

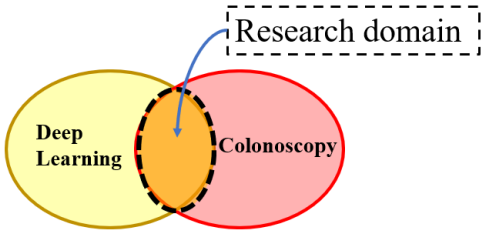


Figure 6 The thesis domain lies between deep learning and colonoscopy.

However, it's important to note that this area of study is currently in its early stages for the following reasons. First, the vast majority of deep learning studies are conducted outside clinical environments, hence, it is anticipated that the current stage of applying deep learning models in the context of colonoscopy is not optimized for real-world clinical applications.

Second, a significant number of studies are evaluated using high-quality selected frames rather than videos which resemble real-clinical environments. As opposed to selected frames, videos possess contextual information that can be utilised to enhance prediction results, though, they also contain non-informative frames which could negatively affect the performance of deep learning models during screening.

Third, a notable absence of benchmark standardization which prevents cross evaluation between different studies. Furthermore, improper dataset preparation for deep learning experiments could result in having exaggerated results. Consequently, the ability of such models to perform well on unseen future data is questionable.

Fourth, a disproportionate portion of studies utilised private datasets due to privacy concerns. Accordingly, ensuring reproducibility and transparency of deep learning models is not achievable. This prevented other researchers from doing thorough investigations to determine their competence in real-clinical environments.

Finally, there are clear discrepancies between natural images (e.g., ImageNet) and colonoscopy images. As opposed to natural images, colonoscopy images do not have specific shape nor defined borders. Furthermore, the elasticity of the inner lining of the colon increases the inter-cluster similarities which make it difficult to identify unhealthy tissues from healthy ones. With that in mind, it is not clear if utilising on-the-shelf models, which are originally proposed for non-medical domain, would be effective on colonoscopy images. Accordingly, new designs should be proposed to target domain-specific tasks.

Given the aforementioned drawbacks, the aim and objectives of this thesis are designed to bridge the gaps that presently obstruct the integration of deep learning models into clinical settings. As a result, the aim of this thesis is formulated as follows:

Aim: Assess the present state of deep learning models used in colonoscopy through a clinical lens, and subsequently propose better evaluation methodologies and deep learning techniques.

With the intention of achieving our defined aim, a series of research questions are crafted. These questions act as the navigation compass behind our exploration and experimentation, ultimately leading to the compilation of this thesis work. The research questions are as follows:

- **Do current evaluation strategies reflect the actual effectiveness of deep learning models?** In other words, are current preclinical experiments on deep learning models signal their actual performance on future real scenarios. Furthermore, are the outcomes presented in the literature accurate or magnified?
- **How to evaluate deep learning models from a clinical perspective?** Precisely, how to utilise currently available datasets to assess deep learning models from a medical viewpoint?

- **How to design a deep learning model that can utilise both spatial and temporal information?** Given that colonoscopists analysing videos, which are "fundamentally" sequences of frames, is there any necessity to design video-based architecture rather than frame-based architecture? Furthermore, how spatial and video features can be merged to boost classification performance?
- **How to design a deep learning framework such that it improves performance of segmentation models on unseen future colonoscopy images?** In other words, which approach should be adopted to enhance the generalisability of segmentation models on colonoscopy images?

To accomplish the thesis aim, a set of objectives were delineated:

Objective 1: To develop a systematic method from a clinical perspective to assess and determine the performance of the existing deep learning models employed in colonoscopy.

Objective 2: To develop a novel deep learning architecture that can exploit spatial as well as temporal information within colonoscopy videos to enhance its efficiency in real-clinical environment.

Objective 3: To develop a novel framework to enable segmentation models to acquire invariant properties by utilising a supervised learning method which aim at enhancing their generalisability capabilities on future unseen samples.

The upcoming chapters are crafted to address the outlined aim, objectives, and questions. Chapter 2 is considered the starting point in which an extensive overview of existing deep learning techniques employed in the broader context of the digestive system is conducted, with a focused exploration on the colon. The goal is to investigate the current evaluation strategies in the literature and accordingly propose an evaluation approach from a clinical perspective (i.e., Objective 1). Accordingly, the proposed evaluation approach is applied in both Chapter 3 and Chapter 4 to contrast the performance of the proposed deep learning model against the current state-of-the-art models.

Moving forwards, Chapter 3 and Chapter 4 delve into two distinct yet interconnected colonoscopy tasks that address Objective 2 and Objective 3, respectively. These tasks are automatic bowel preparation and automatic polyp segmentation. The clinical necessity of bowel preparation as a pivotal preprocessing stage for successful screenings is paramount. Inadequate preparation could lead to the obscuring of polyps by stool and bowel residuals, rendering them undetectable to endoscopists. Failure to identify polyps could adversely impact mortality rates.

In Chapter 3, entire video samples are treated as input. This way, both spatial and temporal information could be exploited to extract contextual information, hence, enhancing deep learning classification accuracy. Accordingly, a video-based deep learning model is proposed to achieve Objective 2.

Chapter 4 attacks the polyp segmentation task in which datasets were collected from different medical centres. Accordingly, the generalisability of deep learning models is investigated. Furthermore, in Chapter 4, a deep learning framework which has a better generalisability than the state-of-the-art models is proposed which fulfils Objective 3. The proposed framework consists of image-to-image transformation unit and a segmentation model. The transformation unit manipulates input images' texture, then they are delivered to the segmentation unit during the training phase. This approach acquired the segmentation model texture invariant properties; hence, it enhances the generalisability performance.

1.4 Thesis Roadmap

The recommended sequence for navigating this thesis is depicted in Figure 7. Chapter 1 establishes an introductory foundation, outlining our motivation and articulating the aim of the study. Subsequently, the literature review can be engaged with, building upon the terminologies introduced in Chapter 1. At this juncture, readers may delve into either Chapter 3 or Chapter 4. Finally, the conclusion of this thesis can be explored in Chapter 5. A detailed information about each chapter is outlined as follows.

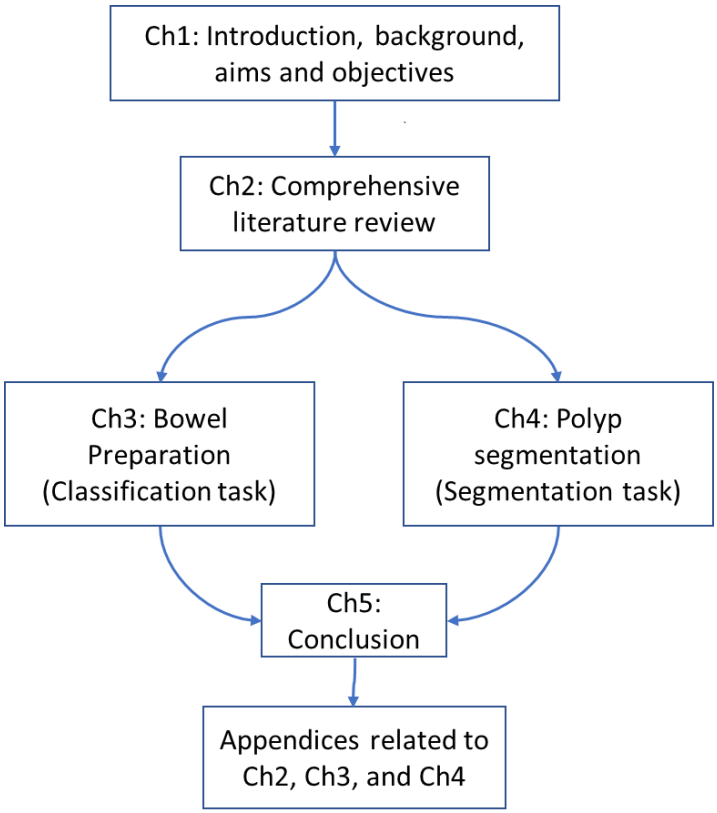


Figure 7 Overview of thesis suggested reading's roadmap.

Chapter 1 serves as an introductory chapter, consisting of general subject matter exploration, consisting of four sections designed to facilitate comprehension of the thesis. The first section introduces commonly referenced technical and medical terminologies within the thesis. The second section discusses challenges in the application of deep learning in colonoscopy. The third section outlines the thesis scope, objectives, aim, and research questions. The fourth section (i.e., this section) illustrates the general outline of the entire thesis. Last section provides a summary of the entire chapter.

Chapter 2 encompasses six sections, in which the first four sections dedicated for investigating the current literature concerning the application of deep learning on the gastrointestinal tract (GI) with focus on colonoscopy. The initial two sections within the chapter explore deep learning methodologies for both upper and lower endoscopy, respectively. Meanwhile, the third and fourth sections are exclusively devoted to investigating the application of deep learning techniques in the realm of colonoscopy. These specific sections of the literature review chapter correspond directly to Chapter 3 and Chapter 4, respectively. Meanwhile section 5 in Chapter 2 wraps up the gaps in the GI literature. Finally, a summary section is provided.

Chapter 3 and Chapter 4 represent the core segments of this thesis, where they delve into a comprehensive empirical exploration of existing voids within the literature. Subsequently, our proposed methodology is presented and conduct a thorough comparison between our model and the state-of-the-art deep learning approaches.

Chapter 5 serves as the conclusion chapter of the thesis, encapsulating the overall findings. The appendix, situated at the end of the document, encompasses comprehensive results from selected experiments and a concise summary of the literature review in a tabular format. Furthermore, observations which are not directly related to the thesis's arguments are provided. The inclusion of detailed results in the appendix is primarily aimed at promoting reproducibility and indorsing ease of comparison with fellow researchers.

1.5 Summary

In this chapter, an introduction is given to illustrate boundaries and domain of this thesis. A technical and background introduction were discussed. An introduction to Gastrointestinal tract (GI) (i.e., digestive system) and the used of different endoscopy were deliberated such as esophagoscopy, gastroscopy, proctoscopy, and colonoscopy. Accordingly, the thesis revolved around deep learning applied to colonoscopy. From a use case point of view, any computer system or model applied to GI will be under one or multiple of the following categories, namely, Computer-aided detection (CADe), Computer-aided diagnosis (CADx), and Computer-aided monitoring (CADm) system. The components of CAD systems can be deep learning model, machine learning model, or any other statistical models. However, the applications of deep learning models can be categorized according to their functionality as a classification model, a detection model, or a segmentation model.

Challenges of deep learning model was also briefed in this chapter. The current state of deep learning models is still considered to be pre-clinical due to several factors including but not limited to, evaluating using high quality handy-picked images, a notable absence of standardized benchmarks, and limited research reproducibility. Furthermore, most published work conducted outside clinical environments.

All the latter factors imposed/enlarged the gap between real-clinical environments and deep learning applications. Consequently, the aim is to evaluate current deep learning methods from a clinical perspective and accordingly propose evaluation strategies and efficient deep learning models. The aim is translated into three objectives and, accordingly, questions were posed to gauge this study. The first objective is to develop an evaluation method to overcome overestimations that may arise due to similarity between training and validation data. The second objective is to design video-based model to utilise colonoscopy videos which ensemble real-clinical situations. The final objective is to enhance generalisability of deep learning models so that it performs better on unseen test data.

To facilitate reading this thesis, a roadmap is provided before this section. This chapter is considered the starting point since it lays out all the background needed to delve into the thesis. Then the second chapter provide literature for the rest of the thesis. Chapter 3 and Chapter 4 can be read in any orders, meanwhile, the last chapter conclude the thesis.

Chapter 2

Literature Review

A thorough presentation of applications of deep learning on endoscopy is presented in this chapter. Accordingly, this chapter is divided into five sections to cover all aspects of this domain. The first two sections devoted towards discussing current deep learning methods applied to upper endoscopy and lower endoscopy. Meanwhile, the third and fourth sections deeply discussed methods applied to colonoscopy which are directly related to the upcoming two chapters, namely, Chapter 3 and Chapter 4. The fifth section wrap up current gaps in the literature. Finally, a summary is presented to conclude this chapter. Moreover, selected studies related to digestive system are summarized in Appendix A.

2.1 Deep learning applied in upper endoscopy

Depending on the targeted region within the upper endoscopy, the application of deep learning (DL) can be categorized into two main streams, namely, DL for esophageal precancerous lesions [40]–[43] and DL for gastric precancerous lesions [44]–[47]. It is worth noting that esophageal squamous dysplasia and Barrett's oesophagus are considered as precancerous states for esophageal squamous cell carcinoma and esophageal adenocarcinoma, respectively. Meanwhile, Helicobacter pylori, atrophic gastritis, gastric intestinal metaplasia, and gastric dysplasia are considered to be precancerous states for gastric cancer [48]. It is important to inspect these precancerous states as early as possible to prevent affected areas from developing into cancerous states.

Most published papers on Barrett's oesophagus analysis have been using mainly hand-crafted features, thus, [49] proposed a computer-aided diagnosis (CAD) system based on deep learning. For training, small patches were generated from endoscopic colour images and augmented to increase the number of samples. Rotation, translation, mirroring along the horizontal and vertical axis, contrast, brightness hue and saturation jittering were used randomly as augmentation methods. The goal is to classify image patches into either esophageal carcinoma (i.e. cancer) or Barrett's oesophagus (i.e. high-grade inflammation). Those classified small patches were then used to delineate/segment object of interest within images. ResNet [7] was used as CNN classifier with two private datasets namely, Augsburg and MICCAI.

Groof and his colleges developed a computer aided system based on deep learning and they tested it during endoscopic screening [41]. They targeted 10 patients with Barrett's oesophagus and other 10 patients without Barrett's. They developed Unet-based architecture with ResNet as encoder to produce both classification and segmentation information. They demonstrated high results in terms of recall, specificity, and accuracy with values of 91%, 89%, and 90%

respectively. The system does not utilise temporal information within video as opposed to the work of [42]. In fact, the first attempt to utilise temporal frames to detect abnormalities in oesophagus was proposed by [42]. The model was proposed to discriminate and detect four categories within the videos frames, namely, 1) Normal, 2) Barrett's Oesophagus (pre-cancer), 3) Esophageal Adenocarcinoma (cancer) and 4) Squamous Cell Carcinoma (cancer). Inspired by [50] they propose a 3D Sequential Dense-Convlstm Faster R-CNN to detect esophageal abnormalities from endoscopic videos. The proposed model is further enhanced by employing a post-processing method named Frame Search Conditional Random Field (FS-CRF). The purpose of FS-CRF is to recover the missing regions in neighbourhood frames and to remove false positives within the same clip. Simply stated, for a frame in a video the FS-CRF method search 15 frames forwards and 15 frames backwards to compensate the missing detection result using CRF [51]. If FS-CRF method didn't find a detection within 15 frames, it assumes that the current frame is normal, and it labelled as false positive sample. The proposed model is able to discriminate and locate abnormalities within oesophagus videos with 93.2% F-score. However, the proposed model is computation-intensive given the fact that a 3D-CNN alone demands high computational complexity [30]. Furthermore, the proposed post processing method (FS-CRF) requires searching frames forwards and backwards which is not possible for real-time environment. Without FS-CRF the proposed model achieved F-score of 88.9%. Therefore, the proposed framework by [42] as it stands is not applicable for real-time clinical practice. However, it may be suitable for applications that don't demand real-time detection such as Wireless Capsule Endoscopy WCE videos [23].

DL for gastric precancerous lesions [45]–[47], [52]. The works of [52] and [44] are considered one of the earliest studies that utilised convolutional neural network CNN to classify Helicobacter pylori. Both work utilised GoogLeNet [53] model on privately collected datasets. The initial results for both works are impressive, yet, given that they utilised private data, it is hard to conclude regards their effectiveness in real-clinical scenarios. On the other hand, Zhang and his team compared their CNN model against human experts for atrophic gastritis classification task [45]. Interestingly enough, their model which is based on DenseNet [54] overcome human experts.

Notable efforts were made to pool all papers related to upper GI diseases in order to estimate its potential clinical value [55]. They pooled 1678 articles and then narrow it down through different exclusion and inclusion criteria to 36 articles. On average, 90% accuracy were achieved across spectrum of different upper GI precancerous diseases [55]. However according to [55], the main drawback of those analysed studies is that they were performed in an experimental settings rather than real-clinical environment in which interaction between AI and the endoscopists is an important aspect of the inspection process. Furthermore, very few studies included videos which are considered to be closer to everyday clinical practice [55].

2.2 Deep learning applied in lower endoscopy

Miss rate of polyp/adenoma detection in colonoscopy is considered one of the factors that limit mortality [27]. Hence, CAD systems have been proposed to increase the detection rate during screening. However, the actual effectiveness of those CAD systems in real-time situation are yet to be investigated. The recent development of deep learning frameworks facilitates researchers to initiate studies that address computer aided CAD systems. In fact, a decent work have been conducted to address deep learning methods applied to detect and segment colorectal polyps [56]. The authors adopted PRISMA recommendations [57] to construct their systematic literature review. Out of 1332 studies, only 35 full texts were eligible for reviewing. The authors noted an increase trend towards the usage of deep learning for polyp classification, detection and segmentation since 2016 [56]. However, still it is not confirmed whether deep learning models would reduce the incidence and mortality rate of colorectal cancer [58]. Moreover, [58] suggested to test deep learning models in the context of less than optimal settings rather than testing on high-quality images.

Nevertheless, there are few attempts to study such systems in real-clinical environments. One of such attempt was conducted by [27]. They used their previously proposed system [29] to assess the effect of the CAD on colorectal adenoma detection rate (ADR). Interestingly, they involved endoscopists having various screening experience levels in the experiments to contrast the performance with and without CAD involvement. They concluded that CAD system increased adenoma detection rate due to an increase in diminutive adenomas which are considered to be small and portrayed less risk for malignancy. On the other hand, researcher team reported that the used system may have limited capacity to assist the endoscopist in detection more adenomas in the cecum and ascending colon due to the higher instability of the colonoscopy in those areas [27].

In order for a system to be useful in real clinical practice, it should be responsive and give real-time results. Accordingly, a recent paper has been published recently that review all types of deep learning models employed in localising and classifying colorectal polyps [59]. They concluded that only 7 studies reported that they considered real-time for polyp localisation and only one study for polyp classification [59]. Furthermore, most of the published work in the polyp analysis domain used private datasets which prevents reproducibility and common ground in comparing different deep learning models [59].

In general, studies in lower endoscopy (i.e., colonoscopy) can be categorized into mainly three types [60], including, a) computer-aided detection (CADe), b) computer-aided diagnosis (CADx), and c) computer-aided monitoring (CADm). More discussion can be found in section 1.1.2. CADe and CADx are applications related to localising abnormalities (i.e., segmentation or boundary box detection) and classifying abnormalities, respectively. Meanwhile, CADm

is any system that provide a secondary support during or after the screening such as polyp size calculation [61], withdrawal time monitoring [62], and bowel preparation assessment [31]. Both CADe and CADx applications constitute most of studies in the colonoscopy literature. Furthermore, polyps are the main lesion type in colonoscopy literature due to the availability of public datasets. The majority of CADx deep learning applied to colonoscopy are utilised to classify cancerous and non-cancerous polyps [60] , though, other abnormalities studies exist such as Crohn's [63] and ulcerative colitis [64]. CADe based on deep learning models are thoroughly discussed in section 2.4.

In [63], on the shelf deep learning model (i.e., pretrained Xception [65]) was utilised to predict Crohn's disease on Wireless Capsule Endoscopy WCE images [63]. They conducted two experiments: 1) entire dataset from 50 patients are mixed and randomly divided to create 5-fold cross validation, 2) leave one-patient-out cross validation. Interestingly, the model achieved on experiment 1 accuracies range from 95.4% to 96.7%, meanwhile, accuracies for experiment 2 range from 73.7% to 98.2%. This variability of experiment 2 (i.e., leave one-patient-out cross validation) signifies concerns regards the generalisability of such models in real clinical environments.

To attack the problem of ulcerative colitis classification, Efficient Attention Mechanism Network (EAM-Net) which combines the efficient channel attention network and spatial attention module is proposed [64]. First, the features are extracted by DenseNet [54] model, and the output is divided into two deep learning units, including recurrent neural networks (RNN) and EAM-Net module to generate attention maps. Both generated feature maps are utilised to produce final classification prediction. The proposed model achieved on average 86% F1-score on two private datasets. However, it is worth noting that they refine their dataset by excluding unclear images with stool, blur, or halos. Accordingly, a drop in performance is anticipated if the system is tested in real clinical settings.

2.3 Automatic Bowel Preparation Assessment (Classification)

There have been multiple proposals for automatically determining the degree of bowel cleansing, as in [36]–[39], [66]–[68]. These methods can be classified into two categories based on their approach: a) conventional machine learning or image processing methods such as those described in [37]–[39], and b) deep learning methods as seen in [36], [39], [66]–[68]. While some of the published work utilised private datasets, others utilised a publicly available dataset known as Nerthus [39]. It is worth pointing out that some studies used the Nerthus dataset for purposes other than estimating bowel preparation such as increasing the number of training images to address other tasks such as disease detection in gastrointestinal tract [69], [70]. Since the main goal is to evaluate the clarity of the bowel such papers as in [69], [70] are excluded from our literature review.

In [37], Support Vector Machine (SVM) was employed to segment stool region in images. They extracted colour features from image blocks and fed them to the SVM to produce a binary mask that indicates stool regions. Despite achieving high sensitivity of 99.25%, the model's performance significantly declined when presented with new images [38]. Additionally, the dataset used in this study is private, which hinders reproducibility.

As opposed to [37], the works in [38] targeted pixels' colour rather than blocks' features to identify the stool region. Initially, a 3D-space was created, where red, green, and blue channels were considered as coordinates of the 3D-space, and all stool pixels were projected in this space. The space was then divided into equally spaced 256 planes along the red axis and only planes with stool pixels were selected, and only planes with stool pixels were selected. Consequently, each plane contained a projection of stool pixels at the corresponding location, and these planes were treated as a 2D classifier at the relevant location. A pixel-level classification was carried out on the validation images, achieving 92.9% and 95% for sensitivity and specificity, respectively. However, this proposed method may not work effectively in practice as the colour of stool and mucosa pixels can appear similar due to variations in light, field of view, bubbles, water, and residual liquid. Additionally, mucosa with a thin layer of semi-transparent liquid stool was not included in the training set [38]. Therefore, the generalisability of this method under various real-life situations are questionable.

A recent research article has been published that provides a bowel preparation score every 30 seconds during the withdrawal phase of colonoscopy and shows the cumulative ratio of frames for each score [36]. However, the authors did not provide any details about the model they used except that they used DenseNet [54] with/without transfer learning. Even though this work was published in 2020, the authors did not employ any public datasets and instead they used a private dataset of colonoscopy videos. Furthermore, they treated as separate images, as a result, they did not utilise the temporal information found in videos

(i.e., consecutive frames). The proposed model achieved an accuracy of 93.33% on 120 images (2.191% of the entire dataset size), while achieving an accuracy of 89.04% when tested on 20 colonoscopy videos. However, the accuracy dropped to 80% when images with bubbles were included in the experiments. According to them, the proposed system outperformed endoscopists of various experience levels.

Previously discussed research utilised proprietary datasets, making it impossible to reproduce and analyse their findings. Moreover, these private datasets were composed of high-quality, hand-selected images that do not accurately represent real-world scenarios.

A group of researchers were motivated to address the lack of a public bowel preparation dataset and thus released a dataset called Nerthus [39]. The owner of the Nerthus dataset conducted preliminary experiments to establish baseline performance using various models. Two primary approaches were employed: a) classification using hand-crafted global features (GF) (e.g., colour layout, edge histogram, auto colour correlogram), b) deep learning convolutional neural networks (CNN) with/without transfer learning. The highest F1-score=89.9% achieved by GF with Logistic Model Tree method.

Deep learning methods were utilised in [66], [67] to evaluate bowel cleansing using Nerthus dataset. In [67], four convolution blocks were used, each consisting of a convolution layer followed by rectified linear unit (ReLU), batch normalisation, and max pooling. Meanwhile, [66] utilised a pre-trained ResNet50 [7], a relational mapping [71], Long Short-Term Memory (LSTM) [72], and three fully-connected layers. The ResNet50 was only used to extract representative features, so all its weights were frozen during training. The relational map "RN" was proposed to map the correspondences between two distinct feature streams (i.e., feature maps from shallow and deep layers).

Both works [66] and [67] achieved high validation accuracy of 100% and 97.7%, respectively. In [68], the Nerthus dataset was used to test a proposed model consisting of a pre-trained ResNet50 attached to a Bayesian neural network [73] to authenticate the prediction accuracy. Similar to the previous methods, [68] achieved a validation accuracy of 100%. However, these results are overestimation due to having similar data distribution in both training and validation as will be demonstrated in the next paragraph.

The papers that used the public Nerthus dataset attained a high validation accuracy. However, it is found in this thesis that the accuracy was not a result of the model's architecture, but rather due to mishandling of the video dataset. The frames of all videos were assumed to be "independent" images, which is problematic when randomly dividing them into training and validation datasets. Due to the high level of similarities between consecutive frames of a video, it is highly likely that the validation dataset would contain images resembling many in the training dataset. This would lead to a trained model being tested on images that it has already "seen" or seen very similarly before, resulting in artificially high testing accuracy. However, this is not reflective of a real-life clinical

environment where none of the frames in a colonoscopy video would have been previously encountered by the model.

Accordingly, it is crucial to ensure that the training frames and validation frames used for evaluating models are entirely distinct and selected from different distributions to obtain accurate evaluations. Otherwise, the model's performance would be overestimated. This overestimation issue is thoroughly demonstrated later in Chapter 3.

2.4 Automatic Polyp Delineation (Semantic Segmentation)

Colorectal cancer is the third most prominent contributor to cancer-related fatalities globally, with an alarming mortality rate of approximately 51% [74]. Overlooked polyps can lead to their transformation into cancerous polyps. To address this issue, numerous public polyp segmentation datasets have been created, allowing researchers to explore the problem using deep learning models. Due to the variety of models utilised, this section has been divided into subsections, each focusing on a specific architectural type.

2.4.1 Fully Convolutional and Encoder-Decoder based models

Fully Convolutional Networks, commonly known as FCNs [75], represent a segmentation model that relies exclusively on convolution, pooling, and up-sampling layers, as seen in Figure 8-(a). Unlike traditional Convolutional Neural Networks (CNNs), FCNs do not incorporate fully-connected layers (dense layers). This unique characteristic allows FCNs to be adaptable to any input resolution [75]. Accordingly, several works investigated the usability of such architecture in polyp segmentation [76]–[79].

Three different FCN-based architectures were trained and fine-tuned for polyp segmentation [77], whereas in [78] multistep were used including region proposal generation using FCN followed by spatial features extracting and utilising forest classifier for the refinement phase. Similar approach is adopted by [76] in which patch selection while training FCN is employed followed by applying Otsu thresholding for accurate polyp localisation.

In general, FCN is known to produce a coarse segmentation output due to its multiple pooling operations which yield a condense feature map. The feature map is then up-sampled only once to produce a segmentation mask. Accordingly, a progressive up-sampling architecture is proposed including encoder-decoder and Unet architecture [8], as seen in Figure 8-(b) and Figure 8-(c). Originally, Unet architecture was primarily designed for biomedical image segmentation tasks. Recently, many variants of Unet-based architecture have been proposed for several segmentation tasks such as lungs segmentation [80], [81], liver tumour segmentation [82], and cell segmentation [83].

Nevertheless, different flavours of Unet were proposed for polyp segmentation task [84]–[87]. For instance, multiple deep encoder-decoder networks were proposed by [84] to contextual information, meanwhile in [85] double Unet was proposed to learn rich information. Mahmud et al [86] demonstrated that a better feature representations can be achieved by utilising dilated inception blocks. Meanwhile, a lightweight encoder-decoder was proposed by [87] which is capable of accurately segment polyps with 86 frame/second.

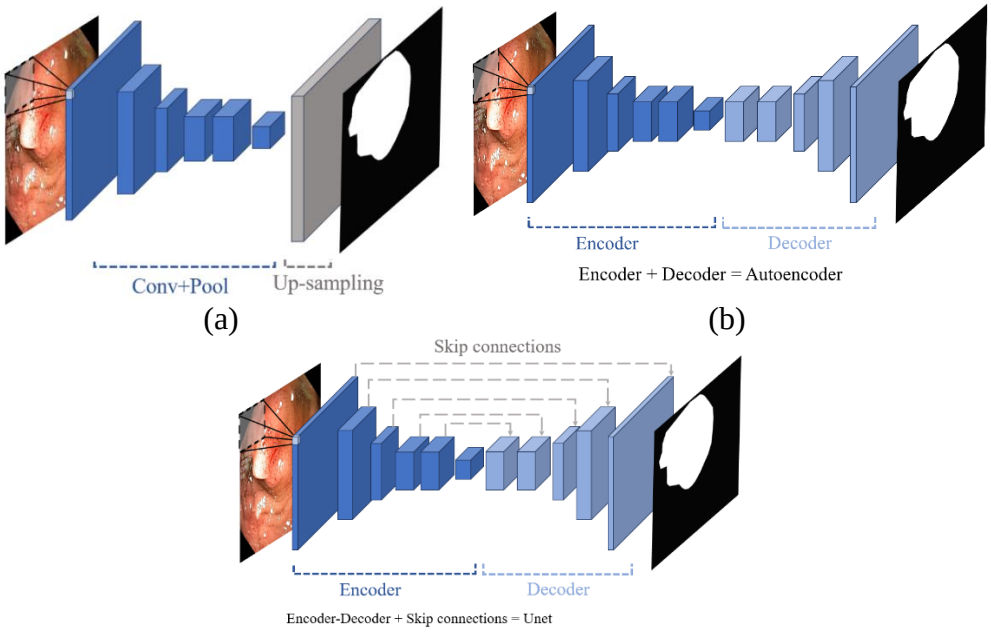


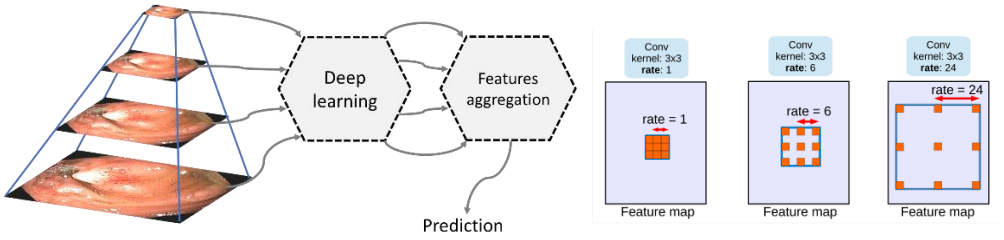
Figure 8 Three conventional types of segmentation models, (a) Fully Convolutional Network FCN [75], (b) Encoder-Decoder network [88], and (c) Autoencoder with skip connections (i.e., Unet [8]).

2.4.2 Pyramid-based models and Dilated convolution based models

Pyramid representation is a type of multi-scale representation developed by signal/image processing communities in which an image is subject to consecutive down sampling. This process results in having a pyramid of stacked images in which each layer in the pyramid has a different scale, as shown in Figure 9-(a). This method enables deep learning models to extract global and local information which results in having better feature representations.

Like the pyramid representation method, dilated convolution is proposed to attack multi-resolution object detection. In contrast to pyramid method, dilated convolution expands convolution kernels to enlarge the field of view of filters to embed multi-scale context [89], as depicted in Figure 9-(b). It is worth mentioning that pyramid and dilated convolution methods are not mutually exclusive, hence they can be applied at the same time.

Pyramid method were utilised for polyp segmentations as in [90] and [91]. In [90] a model called PLPNet was proposed. PLPNet consists of two parts, including feature representation and segmentation (i.e., FCN model) units. Training PLPNet is conducted in two steps. In the first step, the feature representation unit is trained to generate rich features. After that its weights are frozen and the segmentation unit is trained to generate polyp masks. The proposed model performed extraordinarily well on CVC-ClinicDB [92]. Meanwhile in SegNet [88] and U-Net were used to create ensemble deep learning model. To unify the output of the two models, a voting method was utilised.



(a) (b)

Figure 9 (a) Pyramid features representations and (b) Dilated convolution. Image (b) is courtesy of [89].

Dilated convolution has been utilised in various polyp segmentation studies [93]–[96]. Both works in [93], [94] utilised Unet architecture as the backbone architecture, though, the latter utilised attention and residual block in the decoder. Interestingly, both methods achieved a high F1-score of ~96% on CVC-ClinicDB [92]. Given that CVC-ClinicDB consists of series of similar frames, the achieved results may indicate an overestimation due to arbitrary splitting the dataset into training and validation set. This issue is empirically demonstrated in section 4.3.2. On the other hand, DeeplabV3 [89] architecture is the base model of the proposed model in [95], [96]. The DeeplabV3 model combines both pyramid and dilated convolution to enrich feature maps representation, hence, achieving better segmentation results. In [95] Long Short-Term Memory (LSTM) [72] is used to interconnect feature maps from within different blocks of DeeplabV3. Meanwhile in [96], a collective of several DeeplabV3 models is used in conjunction of a pyramid of multi-scale versions of the input images. The work in [95] utilised CVC-ClinicDB for training and validation and reported achieving 93.21% mean Intersection over Union (mIOU) even though the number of images in the training set is less than validation set (i.e., training and validation sets consists of 267 and 345 images, respectively). Again, the achieved results could be exaggerated given the high similarities between the training and validation set (more discussion in section 4.3.2).

2.4.3 Self-attention based models

Self-attention and Transformer-based architecture have been recently adopted in colonoscopy application domain after their prevalent success in language tasks [97]–[101]. The distinction between Transformer and self-attention is that the latter is the basic building block of the former. Added to that, Transformer utilises other learnable components such as input/output embedding , positional encoding and multi-head attention [97].

ACSNet [98] and CaraNet [99] are two deep learning models that utilised attention-based layers. ACSNet consists of Local Context Attention (LCA) module, Global Context module (GCM) and Adaptive Selection Module (ASM). The purpose of ACSNet is to enhance polyp segmentation capabilities on various sizes [98]. On the other hand, CaraNet employed a proposed Context Axial Reverse Attention Network (CaraNet) to improve the segmentation performance on small objects [99]. Such models which are heavily dependent on attention mechanism may easily overfit small training set, hence, it would affect their generalisability performance. This overfitting issue is empirically demonstrated in section 4.3.5.

An encoder-decoder based architecture embedded with attention blocks on the decoder part was proposed in [100]. The utilisation of attention method here is to lessen noise and surplus features, hence, enhances retaining desirable contextual feature relationships. Kvasir-SEG segmentation dataset [102] was used for training, validation, and testing with ratio of 80%, 10%, and 10% of the entire dataset, respectively. Although they used other non-polyp datasets for experimentation, they didn't utilise any unseen polyp test set to verify the generalisability of the proposed model [100]. Meanwhile, in [101] a mix of two parallel convolutional neural networks CNNs along with a Transformer based unit are fused together to capture global and local polyp features. The proposed model in [101] was trained using 1450 training images partly selected from Kvasir-SEG [102] and CVC-ClinicDB [92] (i.e., 80% reserved for training), meanwhile, they tested their model on the rest of the datasets, including, Kvasir-SEG [102], CVC-ClinicDB [92], ETIS-Larib [103], and CVC-EndoSceneStill [104]. Nevertheless, they achieved high mean Intersection over Union (mIoU) of 87% on Kvasir-SEG and 89.7% on CVC-ClinicDB. However, the generalisability of the model is still unclear given that both dataset is partially seen during training phase.

2.5 Current gaps in the literature

The following arguments have been formulated after conducting an extensive and in-depth examination of the literature. Some of these arguments were empirically demonstrated in later chapters of the thesis (specifically, Chapter 3 and Chapter 4). Others were uncovered while reviewing the literature, as well as through participation in colonoscopy workshops, conferences, and competitions. To streamline the discussion, several related arguments have been consolidated under single subsections, resulting in forming several subsections each discussing a specific issue.

2.5.1 Image-based approaches versus video-based approaches

Colonoscopy images or frames are extracted from colonoscopy videos, and as a result, some published works in literature do not make a clear distinction between the two. Consequently, it has been observed that certain studies have labelled their proposals as being applicable to video colonoscopy, even though they developed and assessed their proposed techniques using handy-picked image datasets rather than video datasets. Unlike deep learning models tailored for images, deep learning architectures created for videos must account for temporal information. As a result, image-based and video-based deep learning models exhibit differences in their architectural design.

Additionally, video-based datasets typically remain untrimmed, encompassing non-informative frames such as those that are blurred, out-of-focus, saturated, or contain mucosa residuals. These frames are commonly encountered in real clinical environments and could potentially have a detrimental impact on the performance of deep learning models when tested under such conditions.

Furthermore, certain diseases manifest exclusively in specific regions within the gastrointestinal tract, such as Barrett's oesophagus, which is localised to the z-line in the distal oesophagus [30]. Consequently, contextual awareness assumes paramount significance. This contextual understanding is attainable through the utilisation of sequence modelling on videos. The absence of such awareness could hinder the performance of deep learning models. Indeed, as it is empirically showcased in Chapter 3, the utilisation of videos as training data markedly improved overall performance, surpassing frame-based models and achieving a state-of-the-art outcome.

Up to this point, the deep learning models that have been adopted to handle temporal frames in the gastrointestinal tract and colonoscopy domain have largely been borrowed from other computer vision domains, which are inherently distinct from the endoscopy domain. For instance, the 3D-ConvLstm architecture was initially introduced for detecting human movements in videos, while recently developed Transformers and self-attention methods were initially designed and evaluated for addressing Natural Language Processing (NLP) tasks. Even though these models can function in the medical domain, crafting specialised architectures tailored to the characteristics of colonoscopy videos would likely boost the performance of such deep learning models.

However, it's worth noting that the emphasis on video-based approaches within the literature remains relatively rudimentary in comparison to frame-based approaches especially in the colonoscopy domain.

2.5.2 Subtle overestimation and absence of unified benchmarks

Distinguishing between frame-based and video-based datasets is not only crucial when designing deep learning models but also during the validation and testing phases. The methodology chosen for splitting video frames into training, validation, and test sets holds paramount importance. Videos often contain a substantial number of duplicate frames. Consequently, if videos are treated as individual, unrelated images and shuffled to form training, validation, and test sets, it becomes inevitable to encounter similar frames across all three sets. In such a scenario, even if a deep learning model has overfitted the training set, it may yield high accuracy when tested on the validation and test sets simply because it has encountered analogous images during training. This ultimately leads to an exaggerated assessment of deep learning models. Indeed, as empirically illustrated in Chapter 3, this form of overestimation undoubtedly widens the disparity between empirical research and clinical feasibility.

That being said, certain image-based datasets, like CVC-ClinicDB and ETIS-LaribPolypDB polyp datasets, can also contain semi-sequenced frames. In these datasets, a sequence of frames is extracted from colonoscopy videos for each polyp. Consequently, each polyp is represented in multiple images, potentially exhibiting variations in lighting and angles.

Failing to account for this during the preparation of training and testing images can lead to the presence of similar datasets in both the training and testing sets. As a result, deep learning models may achieve high performance, which is essentially a consequence of arbitrary dataset partitioning. This subtle overestimation is empirically demonstrated in Chapter 4. Consequently, it's not surprising that significant discrepancies in reported results were observed, where some papers achieved notably higher outcomes on these datasets compared to others. A cursory review could erroneously suggest that achieving high results on such datasets signifies being at the forefront of the field.

Hence, one might choose to personally examine these models to form a definitive judgment regarding their performance or to compare them against a proposed model. Nevertheless, achieving reproducibility in deep learning research is widely considered to be challenging, and at times, even unattainable. This is primarily due to the vast number of hyperparameters and intricate implementation details that are frequently left unreported, all of which significantly influence the final outcomes. Even when the source code is made available, the absence of precise information about the colonoscopy images used for training and testing renders true reproducibility an elusive goal.

Consequently, determining from the literature whether a specific method is indeed state-of-the-art becomes problematic, and comparing performance across various papers becomes an invalid approach. By shuffling the training and validation sets differently, one can produce disparate performance results, an occurrence that are empirically verified in Chapter 4. Accordingly, the lack of unified benchmarks that provide clear delineation between training, validation, and testing datasets hinder the assessment of different models across the literature.

2.5.3 Inadequate, homogeneous, and private datasets

The insufficiency of publicly available datasets is a well-recognized issue within the gastrointestinal tract domain. Indeed, this scarcity of public datasets has been a topic of discussion in numerous survey papers within the literature. Consequently, some researchers resort to utilise their proprietary datasets, while in certain cases, they seek unlabelled data from the internet and proceed to annotate it privately. As an example, there has been a notable absence of a publicly accessible endoscopic dataset for oesophagus lesions thus far. Conversely, the existing public endoscopic datasets are notably limited in quantity and predominantly focused on a few specific lesion types. Ultimately, this shortage hampers progress in research within this domain, significantly impeding the applicability of deep learning in addressing endoscopic challenges.

In contrast to the prevailing trend in the literature, our perspective contends that the advancement of computer-aided systems (CAD) within the gastrointestinal tract and colonoscopy domains is progressing at an inactive pace and it is still distant from practical clinical application. Without access to comprehensive public datasets, CAD systems will continue to linger in the preclinical phase. This predicament arises because deep learning models inherently demand a wealth of data, and consequently, their performance is heavily reliant on the quality and quantity of the data they were trained on. Additionally, the use of private datasets for testing hinders the ability to reproduce and compare the performance of these methods.

2.5.4 Domain-specific transfer learning

Quite a few papers have discussed the juxtaposition of domain-specific transfer learning and transfer learning from natural image datasets like ImageNet. Domain-specific transfer learning involves using an endoscopic dataset to pre-train a deep learning model, as opposed to employing well-established datasets like ImageNet or COCO. Given the fundamental disparities between natural images (e.g., those featuring cats, dogs, trees, and humans) and endoscopic images, there are indications that lean towards the adoption of domain-specific transfer learning rather than the conventional transfer learning approach. However, further investigation is required to validate the utility and applicability of domain-specific transfer learning. Consequently, it remains unconfirmed whether utilising domain-specific transfer learning would indeed

outperform the traditional transfer learning technique which depends on extensive non-medical datasets like ImageNet.

2.5.5 Generalisation challenges in deep learning for colonoscopy

The prevailing practice involves conducting training, validation, and testing using either a single dataset or a combination of datasets that got their frames divided randomly. However, as illustrated in Chapter 3 and Chapter 4, this approach tends to result in an overestimation of the reported results. Unfortunately, due to the current dearth of substantial public datasets, evaluating the generalisability of proposed models becomes a challenging endeavour. Reports within the literature have indicated that deep learning models may experience a decline in performance when assessed on datasets sourced from different hospitals or produced by various endoscope manufacturers. Ultimately, this will have ramifications for the integration of deep learning models into routine clinical practice.

2.6 Summary

A lengthy discussion of deep learning methods applied to colonoscopy was presented in this chapter. First, upper and lower endoscopy literature were discussed in general. Then, two subsections were devoted to deliberate methods directly related to Chapter 3 and Chapter 4, respectively. Finally, current gaps in the literature, that is related to the application of deep learning applied to colonoscopy were discussed. The arguments of current gaps were organized in a sectioned format to facilitate the dissection.

It is noticed that deep learning methods have been recently applied to colonoscopy. Machine learning was extensively used before the era of deep learning. Furthermore, there are similarities between methods used for upper and lower endoscopy. A large amount of deep learning methods applied to colonoscopy are inspired/borrowed from non-medical domains. Accordingly, such models are not tailored for colonoscopy, despite, the discrepancy between non-medical domain and colonoscopy domain.

The usage of private datasets disabled research reproducibility which in return hindered the progress in this domain. Furthermore, current available datasets are low in quantity. Furthermore, images are handy-picked high quality which are far from real clinical-settings which in turn rise questions regards their effectiveness in real-life settings. Lack of proper datasets impose questions regards the generalisability of proposed models in the literature.

All the aforementioned gaps indicates that deep learning applied to colonoscopy is still in the pre-clinical stage. Therefore, more efforts and investigations are mandatory in order to enhance the effectiveness of deep learning method to a point in which they are ready to be used on a daily clinical practice.

Chapter 3

Automatic Bowel Preparation Assessment

3.1 Introduction

Colorectal cancer ranks among the primary reasons for cancer-related deaths. It is acknowledged as the second and third most common cause of cancer fatalities worldwide and in the United States, respectively [105]. Early detection of lesions can considerably increase the chances of survival. Although there are various screening methods available for colon screening, colonoscopy is the preferred technique. This involves using an endoscope to screen the colon, whereby a tube with a camera attached at the end (i.e., endoscope) is inserted into the rectum, enabling a healthcare provider (endoscopist) to view the colon's internal structure.

The quality of colonoscopy screening is crucial in preventing colorectal cancers, and there are several factors used to measure its quality, including withdrawal time, thorough examination of the colon, and the quality of bowel preparation [37], [38]. If the bowel is inadequately prepared, the likelihood of missing polyps or lesions increases. This means that the rate of polyp detection is affected by the degree of cleansing of the colon mucosa. Unfortunately, randomized controlled trials have shown that up to 75% of patients experience inappropriate cleansing [106], which can result in a degradation of the quality of colonoscopy screening [107]. Therefore, bowel preparation is a crucial step before and during colonoscopy screening. As a result, both the American Society for Gastrointestinal Endoscopy (ASGE) and the American College of Gastroenterology (ACG) Taskforce on Quality in Endoscopy recommend reporting the quality of bowel preparation [38].

Authors of [108] have proposed Boston Bowel Preparation Scale (BBPS), a method that quantifies the clarity of the bowel. The BBPS quantization method consists of four degrees of clarity (i.e., “poor”, “fair”, “good” and “excellent”) assigned numbers ranging from 0 (poor) to 3 (excellent). However, the quantification method relies on human experts, which can introduce biases, subjectivity, and errors. Implementing an automated system to assess the clarity of the bowel would eliminate these biases and reduce the burden of reporting the cleansing degree in a daily basis routine. Therefore, an automated system for evaluating the quality of colonoscopy screening, including bowel clarity, would be a valuable tool.

It is important to note that research on automatic evaluation of bowel preparation has been conducted previously [37], [38], [66], [67]. However, the actual performance of these systems in real-life situations are questionable for various reasons. For instance, previous research mostly utilised private datasets, hence preventing research reproducibility. Additionally, these private datasets were filtered to exclude non-informative frames, such as blurry frames, bubbles, out-of-focus frames, and saturated frames. However, such frames are commonly found in real clinical environments and could negatively impact the performance of the proposed system.

Fortunately, only recently has a public dataset (Nerthus) [39] been introduced, which includes short videos containing both informative and non-informative frames. A dataset containing videos is more representative of clinical real-environment than a dataset with handy-picked frames. However, failure to recognize Nerthus as a video dataset could lead to an overestimation of the proposed model's performance. Since videos consist of nearly identical consecutive frames, arbitrary dividing them into training and validation sets can result in high similarity between the two sets. If both sets are drawn from the same distribution, any trivial model would perform well. However, its performance would significantly drop when tested on an unseen dataset, rendering the model less useful. This issue has propagated to several works in literature [39], [66]–[68], and attaining such high performance can discourage further research in automatic bowel preparation evaluation. This has led to a halt in research in this area, despite the severe lack of public datasets in the colonoscopy domain. This problem is elaborated and empirically demonstrated in the subsequent sections.

Section 3.1 provides a concise yet thorough literature review, while section 3.2 introduces the proposed framework and delves into its intuition and analysis. In section 3.3, the experiments and results are discussed, and section 3.4 provides the conclusion of this chapter.

3.2 Methodology

This section will showcase the public video dataset, Nerthus, with examples, and empirically demonstrate the issue that arises when treating the dataset as a collection of independent frames rather than videos. Due to GPU limitations, the proposed model was not able to be trained using entire videos as input. Therefore, a sampling method that enables us to use a shorter version of the input video without losing critical information is presented. Finally, analysis and intuitive discussion of the proposed model are illustrated.

3.2.1 Nerthus Dataset

The Nerthus dataset comprises of videos featuring varying levels of bowel preparation degree [39]. The dataset creators utilised the Boston Bowel Preparation Scale (BBPS) [108] scheme to categorize the videos, which is depicted in Figure 10. The label "class 0" signifies that the bowel is completely covered with solid stool, whereas "class 3" indicates that the bowel is completely clear. There are a total of four classes, as demonstrated in Figure 10. The video durations range from 7 to 15 seconds, with a frame rate of roughly 25 frames per second. Therefore, each video contains 175 to 375 frames, as outlined in Table 1. Additionally, the number of videos in each class is uneven.

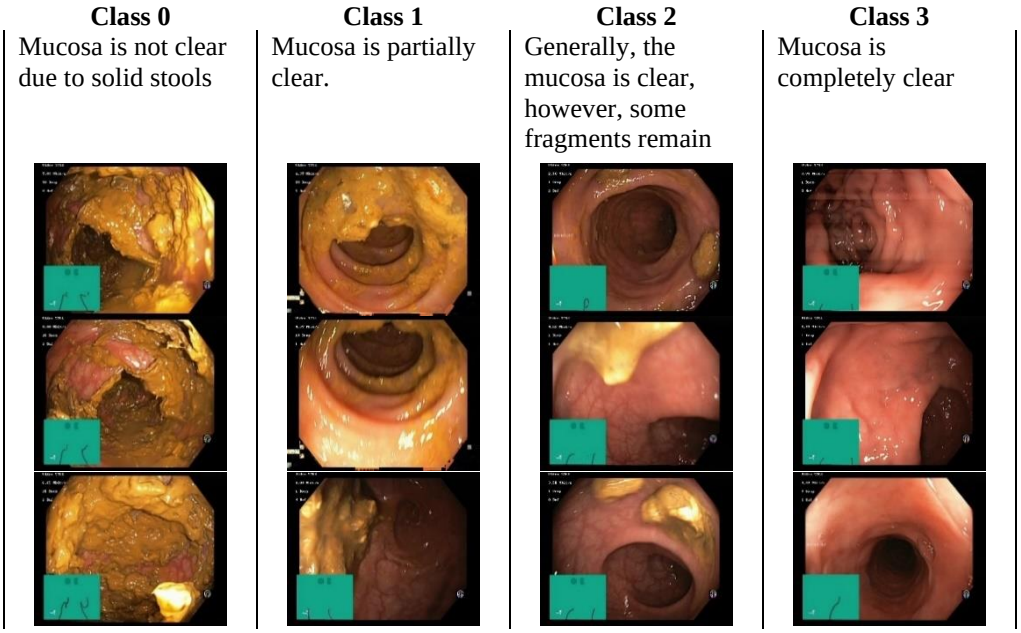


Figure 10. Nerthus dataset labels with a description and examples for each corresponding label.

Table 1 Each class in Nerthus dataset along with their corresponding videos and frames are listed.

	Class 0		**Class 1**									
Video Number	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}	V_{11}	V_{12}
Frames	225	275	275	175	275	325	250	325	275	300	250	250
Total	500		2700									
	Class 2				**Class 3**							
Video Number	V_{13}	V_{14}	V_{15}	V_{16}	V_{17}	V_{18}	V_{19}	V_{20}	V_{21}			
Frames	250	175	375	175	275	250	250	325	250			
Total	975				1350							

It should be taken into consideration that while a video may be classified under a certain category, certain frames within the video may considered non-informative images such as blurred and saturated images or may belong to a different class altogether. For instance, a few frames within a video may visually appear to be categorized as "class 2", even though the entire video is categorized as "class 1" due to the overall appearance of the bowel in that video. This is a typical occurrence as different regions of a bowel may have varying degrees of clarity, which can lead to uncertainty in determining the actual classification.

There are only two videos available for "class 0" which limits the number of folds that can be used as cross-validation sets. As a result, a 2-fold cross-validation approach is employed to evaluate the effectiveness of the proposed model. However, three shuffles of the complete dataset were conducted and, for each shuffle, 2-fold cross-validation were carried out in accordance with the steps outlined in the next section.

3.2.2 Sampling videos and creating cross-validation sets

Because GPUs have a limited capacity, considering complete videos as training batches was not attainable. However, it is known that a video comprises nearly identical consecutive frames, which allowed us to implement a straightforward sampling technique. This method generates smaller, but still indicative, sub-videos for each video in the dataset. This section elaborates on the used sampling method.

The core concept is to extract a single frame from various positions in a video and create a condensed version of that video (referred to as a sample). As the video consists of almost identical consecutive frames, capturing one frame from each location in the video is sufficient to represent the video's overall information. Figure 11 illustrates this sampling technique.

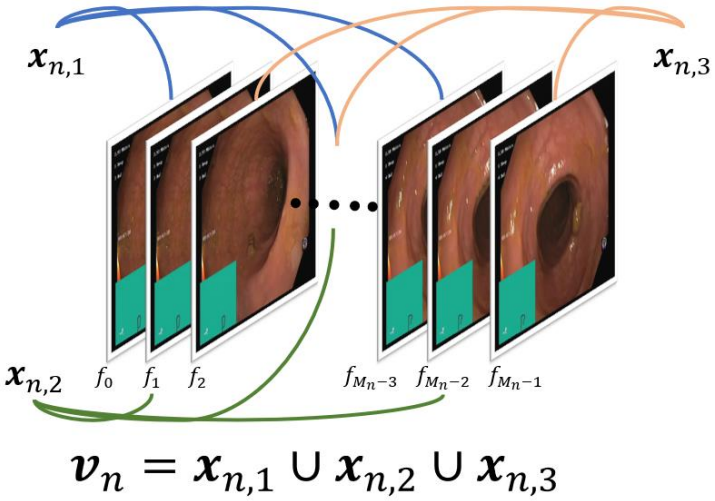


Figure 11 Sampling representative sub videos out of a single video.

Let $V = v_1, \dots, v_N$ represent a set of N videos with corresponding labels $Y = \{y_1, \dots, y_N\}$. Then the training set can be represented as a collection of tuples $T = \{(v_1, y_1), \dots, (v_N, y_N)\}$. Each tuple in the set T consists of an input video v_n and its corresponding label y_n. Furthermore, each video consists of frames $v_n = \{f_0, \dots, f_M\}$. However, because of the limitations of the GPU capacity, these input pairs in T couldn't be used to train the proposed model. To resolve the GPU storage limitation, representative samples (i.e., sub-videos $x_{n,0}, x_{n,1}, \dots$) from each video v_n is proposed using the following equation:

$$x_{n,j} = \left\{ \bigcup_{i}^{s-1} f_{j+d_n \cdot i} \right\}, \qquad j = 0, \dots, d_n - 1 \tag{1}$$

where s is a user-defined variable that determines the size of a sub-video and $d_n = \left\lfloor \frac{M_n}{s} \right\rfloor$ is the effective sampling step for video v_n that has a total number of frames M_n. Note that the created sub-videos $x_{n,j}$ have the following properties:

$$\bigcup_{j} x_{n,j} = v_n \tag{2}$$

$$\bigcap_{j} x_{n,j} = \emptyset \tag{3}$$

$$\left| x_{n,j} \right| = s \tag{4}$$

$$|v_n| = \left| \{x_{n,0}, x_{n,1}, \dots\} \right| = \left\lfloor \frac{M_n}{s} \right\rfloor = d_n \in \mathbb{R} \tag{5}$$

That is, each video v_n will be sampled into $d_n \in \mathbb{R}$ smaller representative videos, each of them would have the same number of frames s . The newly created representative videos $x_{n,j}$ are subset of the original video v_n.

Considering the listed settings (1) to (5), the new training set $\tilde{T} = \{(\boldsymbol{x}_{1,0}, y_1), (\boldsymbol{x}_{1,1}, y_1), \dots, (\boldsymbol{x}_{1,d}, y_1), (\boldsymbol{x}_{2,0}, y_2), \dots\}$. To put it differently, every video from the initial dataset T would be depicted by sub-videos with identical labelling as the original video.

Table 2, Table 3, and Table 4 present the number of newly generated samples and their corresponding videos using the proposed sampling method, forming our new dataset for the training and validation sets. The sub-video size is set to be $s = 25\ frames\ per\ video$ (i.e., by trial and error). Given that the Nerthus dataset contains variable video sizes, 25 frames per video is the greatest common divisor of all the videos in the dataset. This approach ensures that all frames are utilised by the proposed model and enables comparisons between frame-level models and the proposed video-level model.

Table 2 2-fold cross-validation for dataset1. videos with their corresponding samples are listed.

Class	**Dataset 1**					
	Fold1 validation			**Fold2 validation**		
	Video number	***Video samples***	***Total frames***	***Video number***	***Video samples***	***Total frames***
Class 0 (stool)	V_1	9	225	V_2	11	275
Class 1	V_3 V_5 V_6 V_8 V_9	59	1475	V_4 V_7 V_{10} V_{11} V_{12}	49	1225
Class 2	V_{13} V_{16}	17	425	V_{14} V_{15}	22	550
Class 3 (clear)	V_{17} V_{20}	24	600	V_{18} V_{19} V_{21}	30	750
Total	**10**	**109**	**2725**	**11**	**112**	**2800**

Table 3 2-fold cross-validation for dataset2. videos with their corresponding samples are listed.

Class	**Dataset2**					
	Fold1 validation			**Fold2 validation**		
	Video number	***Video samples***	***Total frames***	***Video number***	***Video samples***	***Total frames***
Class 0 (stool)	V_2	11	275	V_1	9	225
Class 1	$V_3 V_4$ V_5 $V_7 V_9$	50	1250	$V_6 V_8$ V_{10} $V_{11} V_{12}$	58	1450
Class 2	$V_{14} V_{15}$	22	550	$V_{13} V_{16}$	17	425
Class 3 (clear)	$V_{17} V_{18}$	21	525	$V_{19} V_{20}$ V_{21}	33	825
Total	**10**	**104**	**2600**	**11**	**117**	**2925**

Table 4 2-fold cross-validation for dataset3. videos with their corresponding samples are listed.

Class	Dataset3					
	Fold1 validation			Fold2 validation		
	Video number	*Video samples*	*Total frames*	*Video number*	*Video samples*	*Total frames*
Class 0 (stool)	V_1	9	225	V_2	11	275
Class 1	V_4 V_6 V_8 V_{10} V_{12}	55	1375	V_3 V_5 V_7 V_9 V_{11}	53	1325
Class 2	V_{15} V_{14}	22	550	V_{16} V_{13}	17	425
Class 3 (clear)	V_{17} V_{21}	21	525	V_{18} V_{19} V_{20}	33	825
Total	**10**	**107**	**2675**	**11**	**114**	**2850**

3.2.3 Proposed model

Colonoscopy videos in Nerthus dataset contain frames that are non-informative, such as those that are blurry, out of focus, or over-saturated, and sometimes even include frames that do not belong to the appropriate category. Nevertheless, there are informative frames within the videos that are nearly identical to each other and can be useful for training purposes. By using these consecutive frames, issues related to non-informative video segments can be addressed, hence, improving the overall performance of the model [30]. To achieve this, a model that incorporate consecutive temporal frames by employing a recurrent neural network RNN model is proposed, namely, Gated Recurrent Unit (GRU) [109].

In addition, utilising non-sequential information by choosing a key frame is proposed. The key frame can indicate the category of the corresponding video. This way, both sequential and non-sequential information in a video can be utilised, as illustrated in Figure 12. Moreover, Figure 13 is presented to aid in illustrating the conceptual framework of the proposed model in relation to the design hierarchy.

The proposed model has mainly four components:

- The base component is an encoder that maps frames to vectors. This could be ResNet50 [7] or any other model such as DenseNet [54] or Inception. Both ResNet50 [7] and VGG11 [110] were used as an encoder for the proposed model and both options were evaluated in the experiments. Furthermore, both backbone models achieved comparable validation accuracy.
- A sequence-based layer that captures temporal information. For this purpose, Gated Recurrent Unit (GRU) [109] is used.
- A none-Sequence based layer that selects a key frame. The proposed None-Sequence layer is called “Multiplexer”. This name indicates the functionality of this layer. Given multiple feature vectors, the Multiplexer would forward a selected feature vector (i.e., key frame’s feature vector). The produced vector by this layer $\boldsymbol{r}_i$ is scaled to match the magnitude of

the one produced by the GRU layer $\boldsymbol{g}$. More details are given in the following section.

- Finally, a fully-connected layer is utilised to extract high-level features from vectors that are produced by both the Sequence and None-Sequence layers (i.e., the GRU and Multiplexer, respectively). SoftMax is then applied to generate probabilities for each class based on the resulting vectors.

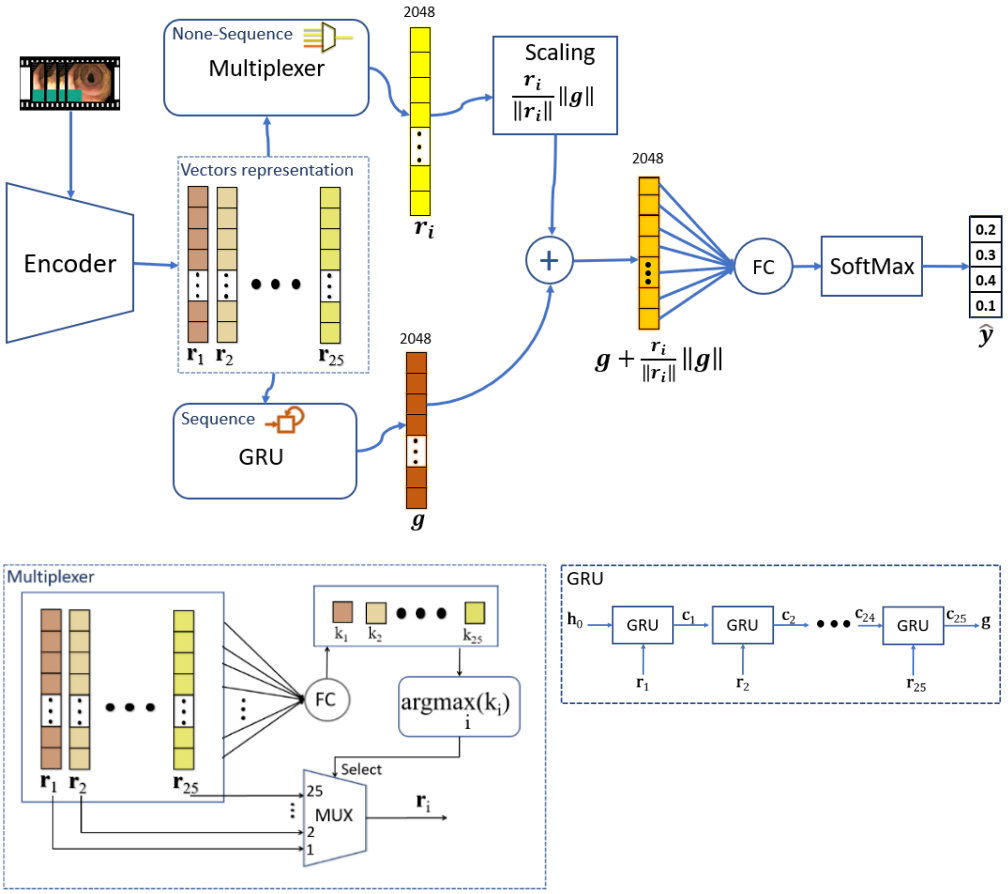


Figure 12. The proposed model has mainly four components: an encoder, GRU, Multiplexer, and fully-connected layer to generate probabilities for each class.

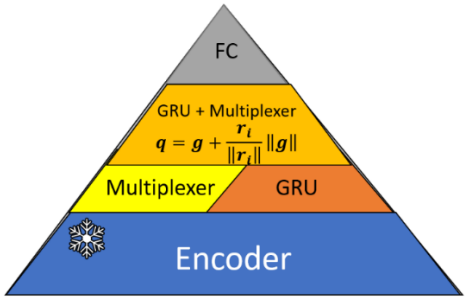


Figure 13. Hierarchal overview of the proposed model.

3.2.4 Intuition of the proposed architecture

Since training input consist of sequences of consecutive frames, it's essential to take into account the temporal information inherent within these sequences. To capture this information, Gated Recurrent Unit GRU [109] is used, which has been shown to be effective in this regard. The resulting feature map vector produced by the GRU layer is denoted as $\boldsymbol{g}$.

By examining Nerthus dataset, it is possible to choose a single frame from each video that can serve as a representation for the entire video's class. To achieve this, a layer named "Multiplexer" is proposed that produces key vectors. These keys are employed to extract a sole feature vector "$\boldsymbol{r}_i$" that corresponds to a specific frame in the given video, as illustrated in Figure 12. This selected feature vector "$\boldsymbol{r}_i$" would then serve as the representative for the whole video.

The objective is to make use of both the sequential data embedded within videos and the key-frame located within each video. The feature vector that encompasses the sequential information is denoted as the vector $\boldsymbol{g} \in \mathbb{R}^{2048}$, whereas the single key-frame features vector is represented by the vector $\boldsymbol{r}_i \in \mathbb{R}^{2048}$, with $i \in \{1,2,\dots,25\}$ indicating the frame position or index. Both $\boldsymbol{g}$ and $\boldsymbol{r}_i$ are vectors of size 2048. An elementary equation is employed to combine the information from both the sequential and non-sequential data (i.e., $\boldsymbol{g}$ and $\boldsymbol{r}_i$):

$$\boldsymbol{q} = \boldsymbol{g} + \boldsymbol{r}_i \tag{6}$$

The vector $\boldsymbol{q}$ (appearing in Equation (6)) signifies a feature vector that corresponds to high-dimensional data, namely a video. Assuming there are two video sets **A** and **B** that belong to different class labels, an effective dimensionality reduction mapping f should possess the following characteristic:

$$d\big(f(\mathbf{A}_i), f(\mathbf{A}_n)\big) < d\left(f(\mathbf{A}_i), f\big(\mathbf{B}_j\big)\right) \tag{7}$$

$$d\left(f\big(\boldsymbol{B}_j\big), f(\boldsymbol{B}_m)\right) < d\left(f(\boldsymbol{A}_i), f\big(\boldsymbol{B}_j\big)\right) \tag{8}$$

In this context, $d(\cdot)$ represents a distance metric and $i, j, m \; and \; n \in \mathbb{N}$. Essentially, an effective projection function should be able to project vectors in a way that preserves the differences and similarities within and between classes (i.e., inter and intra class correlation). Equation (6) is designed to achieve this goal, as depicted in . demonstrates the impact of adding two vectors, $\boldsymbol{g}$ and $\boldsymbol{r}_i$, which alters the direction and magnitude of the resultant vector $\boldsymbol{q}$ and compensates for any faulty mapping produced by either $\boldsymbol{g}$ or $\boldsymbol{r}_i$. If the sequenced layer, GRU, generates two similar feature vectors $\boldsymbol{g}_0$ and $\boldsymbol{g}_1$ that belong to different classes, the Multiplexer layer generates vectors, which redirects the feature vectors to be further apart.

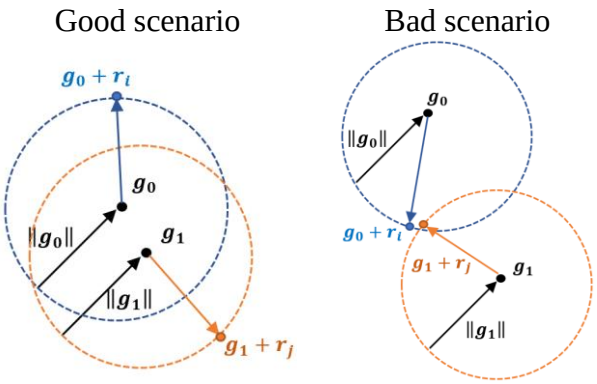


Figure 14. A conceptual view regards adding the two vectors generated by Multiplexer "r" and GRU "g". The dashed circle represents the range of the addition of two vectors given all possible values of "r".

Equations (7) and (8) represent the optimal situation in which the vector $\boldsymbol{r}$ successfully maintains the differences between similar feature vectors, as demonstrated in . However, in the worst-case scenario, $\boldsymbol{r}$ may make two distinct vectors appear more alike. Nevertheless, both the GRU and Multiplexer layers are capable of learning how to produce the corresponding vectors $\boldsymbol{g}$ and $\boldsymbol{r}_i$ during the training phase. The experiment section, specifically section 3.3.5, depicts the visualisation of the produced feature vectors $\boldsymbol{q}$ using two projection techniques namely, principle component analysis PCA [111] and t-distributed stochastic neighbour embedding t-SNE [112].

The GRU layer's feature vector $\boldsymbol{g}$ is regarded as the primary feature vector as it aggregates all frames of a video. On the other hand, the feature vector generated by the Multiplexer $\boldsymbol{r_i}$ is viewed as a supplementary element. Therefore, to utilise the vector $\boldsymbol{r_i}$ as a guide for the vector $\boldsymbol{g}$, it is necessary to regulate the magnitude of $\boldsymbol{r_i}$ which is produced by the Multiplexer layer. This is achieved by normalising the magnitude of $\boldsymbol{r_i}$ to match that of $\boldsymbol{g}$, thereby controlling the impact of the former on the latter. The normalisation is achieved using the following equation:

$$\boldsymbol{q} = \boldsymbol{g} + \frac{\boldsymbol{r_i}}{\|\boldsymbol{r_i}\|}\|\boldsymbol{g}\| \tag{9}$$

Alternatively, instead of using Equation (9), it is possible to normalise both vectors $\boldsymbol{g}$ and $\boldsymbol{r_i}$ so that they have the same magnitude. Nevertheless, this method yielded comparatively inferior outcomes. The impact of this normalisation, as well as the associated results, are deliberated in the experiment section 3.3.6.

3.3 Results and discussions

In this section, the parameters, metrics, and training mechanism used in the proposed model is introduced. Also, a demonstration on how treating a video dataset (Nerthus) as a set of individual images can result in overestimation is discussed. Next, the experimental results of the proposed model are then compared to various state-of-the-art deep learning models, which have been improved through the use of transfer learning. To gain a better understanding of the proposed model's performance, t-distributed stochastic neighbour embedding (t-SNE) [112] and principal component analysis (PCA) [111] are utilised to visualise the feature vectors (i.e. $\boldsymbol{g} + \boldsymbol{r_i}$) generated by the proposed model. Furthermore, a discuss on how selecting different video sizes affects the proposed model's performance will be discussed in a later subsection. At the end of this section, mathematical analyses of the effect of the proposed normalisation (i.e., Equation (9)) on the proposed model will be discussed. Finally, a summary is given to wrap up and highlights core arguments of this chapter.

3.3.1 The used parameters and metrics

PyTorch framework [113] have been adopted to develop the proposed model and experiments. The used GPU are Tesla P100-PCIE and Tesla T4. The hyperparameters used in this experiment are summarized in Table 5.

Table 5 Hyperparameters used for the experiments.

Hyperparameter	**Value**
Epochs	150
Learning rate lr	0.001
Optimizer	Stochastic Gradient Descent SGD
Momentum	0.9
Batch size (for the proposed model)	4 videos (25 frames/video)
Batch size (for SOTA models)	100 frames

The used objective loss is weighted negative log likelihood loss [114]. Given a generated probability matrix $\widehat{\boldsymbol{Y}} \in \mathbb{R}^{N\times 4}$, corresponding target labels $\boldsymbol{y} \in \mathbb{R}^{N}$ $where\ y_n \in \{0,1,2,3\}$, and a batch size N, the loss is defined as follows:

$$l_n(y_n, \widehat{\boldsymbol{Y}}_n) = -w_{y_n} log(\hat{y}_{n,y_n}), \qquad n = 1,2,\dots,N \qquad (10)$$

where w_y is a weight for the corresponding target label y. The weights are employed to mitigate the used unbalanced dataset available. The weights used in all experiments $\boldsymbol{w} = (0.307, 0.159, 0.278, 0.255)$. Class 1 is the dominant class in terms of the total number of images as seen in Table 1. Hence, it assigned the lowest weight (i.e., 0.159). The weights were calculated as follows. First, the total number of images for each class was normalised to be between

[0,1] (i.e., frames = [500, 2700, 975, 1350]/5525 = [0.09, 0.488, 0.176, 0.244]). Then the complement is applied to each element and normalised so that the total sum of all weights is one (i.e., [0.91, 0.512, 0.824, 0.756]/2.96 ≈ [0.307, 0.159, 0.278, 0.255]).

The mean loss is calculated to do backpropagation and update models' weights [114]:

$$L = \sum_{n=1}^{N} \frac{l_n}{\sum_{n=1}^{N} w_{y_n}} \tag{11}$$

Common metrics in the literature were used in this thesis, namely, F1-score, Precision, Recall, and Accuracy. The weighted average is adopted to combine the metrics across all 4 classes.

These metrics, including F1-score, Precision, Recall, and Accuracy, rely on the identification of true positives, true negatives, false positives, and false negatives. Here are the definitions of these sample classifications used in the computation of these metrics:

TP (True Positive): An image or sample that belongs to a specific class (label) and the classification model correctly predicts it as belonging to that class.

TN (True Negative): An image or sample that does not belong to a specific class (label) and the classification model correctly predicts it as not belonging to that class.

FP (False Positive): An image or sample that does not belong to a specific class (label) and the classification model incorrectly predicts it as belonging to that class.

FN (False Negative): An image or sample that belongs to a specific class (label) and the classification model incorrectly predicts it as not belonging to that class.

Accordingly, the definitions of including accuracy, precision, recall, and F1 score as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

$$F1 - score = \frac{2TP}{2TP + FP + FN} \tag{15}$$

3.3.2 Subtle overestimation in the literature

The hierarchy of the Nerthus dataset is shown in Figure 15. There are a total of 21 videos available for training, but the literature considers individual frames as input. Since consecutive frames in a video are nearly identical, randomly assigning images to training and validation set would result in having two similar sets. Accordingly, high validation accuracy and overestimation would be achieved by any model. Even if the videos were divided into halves, shuffled, and split into training and validation sets, the overestimation issue would not be resolved. This is due to having similar frames in both halves, as depicted in Figure 15 -at the sub-video level. This issue has resulted in nearly 100% validation accuracy in previous studies [67], [115], [116]. As far as this thesis reveals, Nerthus dataset has not been treated as a collection of videos in literature.

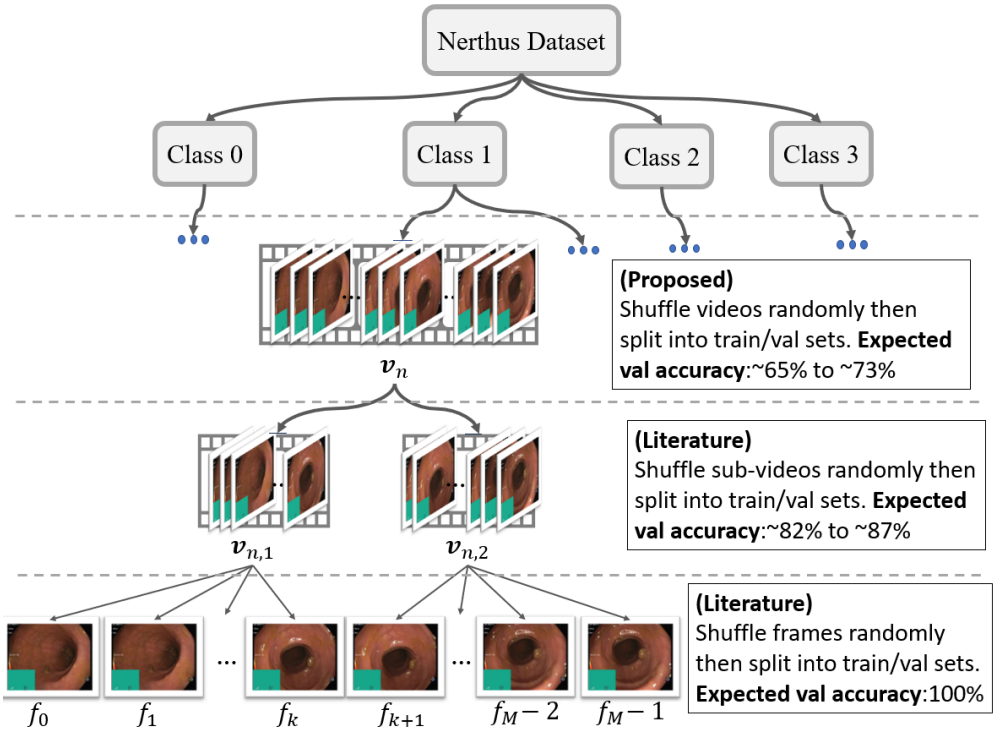


Figure 15 Hierarchy of Nerthus dataset. Nerthus dataset can be viewed as a collection of Videos, Sub-videos, or Frames.

However, considering entire videos as an input/instance is the proper way to overcome overestimation issue. Therefore, training and validation sets should consist of entire videos so that both sets become different. Empirically, experiments at each level were conducted and summarized in Table 6 and Figure 16. Each model in the table was tested by considering Nerthus as a collection of frames, sub-videos (i.e., video divided into half), and videos. A significant drop in performance for each model is noticed when video level is targeted.

To address the issue of overestimation, it is recommended to use entire videos as input during training and validation. This ensures that both sets are distinct. Experimental results, presented in Table 6 and Figure 16, demonstrate that testing each model with Nerthus as a collection of frames, sub-videos (i.e., videos divided in half), and full videos reveals a significant decrease in performance at the video level for all models.

Table 6 Various models are tested with different level configurations. Notice the difference in performance between Frames level and Videos level.

Model	**Validation Accuracy**		
	Frames	***Sub-videos***	***Videos***
ResNet50 [7]	0.9982	0. 8241	0.6837
ViT [117]	0.9982	0. 8212	0.7516
MLP_Mixer [118]	0.9973	0. 8353	0.6462
VGG11 [110]	1	0. 8526	0.7156
InceptionV3 [119]	0.9982	0. 8241	0.6899
DenseNet [54]	0.9991	0. 8673	0.6683

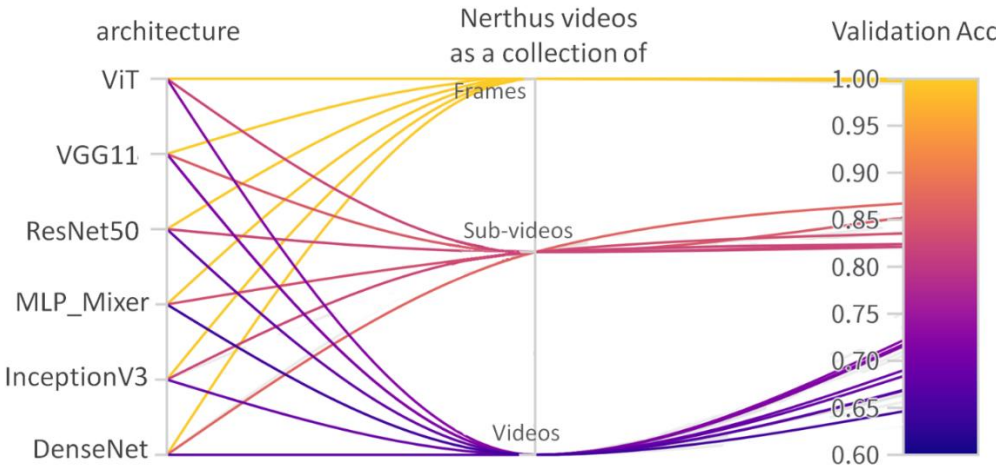


Figure 16 This Parallel Coordinates conveys the information listed in Table 6. Notice the drop in performance between Frame level and Videos level for each model

3.3.3 Comparisons against the state-of-the-art models

To evaluate the performance of the proposed model, the videos were randomized, and a 2-fold cross-validation setup was established for conducting the experiments. This procedure was repeated three times to generate multiple folds, which are presented in Table 2, Table 3, and Table 4. Explicit listing of the video numbers in the aforementioned tables allows other researchers to compare their findings with the proposed model. In this section, the proposed model is compared against the state-of-the-art (SOTA) “frame-level” models to accommodate the literature approach. In addition, the novelty of the proposed model is demonstrated by comparing it with state-of-the-art "video-level" models from non-medical domains. Lastly, the impact of video sample sizes on the proposed model is discussed.

Proposed model against frame-level models

Various models, including both conventional and new ones such as ResNet50 [7] and Vision Transformer (ViT) [117], were tested. The micro average (i.e., weighted average) for each metric across all validation folds can be found in Table 7. Accordingly, a bar chart is provided in Figure 17. To facilitate future benchmarking on this dataset, the specific details of the three shuffled datasets are outlined in Table 2, Table 3, and Table 4. Additionally, the standard deviation is listed to show the variance across the different validation folds. The best two records for each metric results are highlighted. Furthermore, a whisker plot for the F1-score is shown in Figure 18.

On average, the proposed model consistently achieved the highest performance across all metrics. In fact, there exists a significant margin between the proposed model and the next best model in terms of Precision, Recall, and F1-score, with differences of 5.19%, 5.59%, and 6.19% respectively. Based on the findings presented in Table 7 and Figure 17, it can be concluded that the proposed model outperformed state-of-the-art models that utilise frame-level to form classification predictions. Furthermore, it can also be inferred that, contrary to existing literature, employing a video-level model is the correct approach, considering that Nerthus comprises a collection of labelled videos.

The proposed model utilised two different backbone models (i.e., encoders): ResNet50+TL and VGG11+TL. Both encoders were pre-trained on ImageNet and fine-tuned on the Nerthus dataset. As shown in Table 7, the VGG11+TL (i.e., SOTA) model outperformed the ResNet50+TL (i.e., SOTA) model, leading to improved overall results when used for the proposed model. Therefore, the proposed model, utilising VGG11 as the encoder, exhibited better performance compared to the model employing ResNet50. This observation suggests that the choice of the encoder in the proposed model played a role in enhancing the overall performance. Nevertheless, it is important to highlight that substantial improvements were achieved regardless of the specific backbone model chosen, as clearly evident in Figure 17. These improvements can be attributed to the utilisation of temporal and spatial information through the proposed GRU and Multiplexer layers, respectively.

Observations from Table 7 reveal that the proposed model's standard deviation for each metric falls neither at the lowest nor highest end. To visualise the deviation of the F1-score metric, a whisker plot (Figure 18) was generated for comparison. The fine-tuned models, such as ViT+TL, displayed improved results compared to the vanilla models (i.e., ViT without transfer learning). As a result, only the fine-tuned models were selected for the F1-scores comparisons against the proposed model. Notably, the median F1-score of both variants of the proposed model surpassed that of other state-of-the-art models.

Table 7 The average and standard deviation are calculated over all validation folds (i.e., 6 folds in total). The first and second highest results are highlighted.

Models	Precision	Recall	F1-score	Accuracy
InceptionV3 [119]	73.79 ± 5.26	73.03 ± 5.19	71.39 ± 4.81	73.03 ± 5.19
VGG11 [110]	73.61 ± 4.89	70.39 ± 6.14	69.37 ± 5.5	70.39 ± 6.14
DenseNet [54]	69.69 ± 6.51	67.87 ± 6.65	66.28 ± 6.4	67.87 ± 6.65
ViT [117]	71.99 ± 4.77	70.86 ± 4.49	70.22 ± 5.12	70.86 ± 4.49
ResNet50 [7]	68.61 ± 4.52	69.39 ± 6.1	67.18 ± 5.38	69.39 ± 6.1
MLP_Mixer [118]	68.77 ± 6.33	66.02 ± 6.88	65.96 ± 6.65	66.02 ± 6.88
InceptionV3 +TL	75.94 ± 5.13	72.71 ± 4.41	69.42 ± 5.78	72.71 ± 4.41
VGG11 +TL	80.88 ± 2.88	81.3 ± 2.48	79.52 ± 2.92	81.3 ± 2.48
DenseNet +TL	79.63 ± 3.23	79.05 ± 3.19	77.66 ± 3.16	79.05 ± 3.19
ViT +TL	84.35 ± 2.55	82.18 ± 3.51	81.94 ± 2.65	82.18 ± 3.51
ResNet50 +TL	79.31 ± 4.23	79.22 ± 2.08	76.35 ± 2.18	79.35 ± 2.12
MLP_Mixer +TL	81.99 ± 2.77	79.57 ± 5.33	77.72 ± 6.21	79.57 ± 5.33
Proposed (Encoder: ResNet50)	**87.94** ± 4.83	**86.28** ± 6.28	**86.06** ± 5.92	**86.28** ± 6.28
Proposed (Encoder: VGG11)	**91.74** ± 3.95	**89.68** ± 4.73	**89.41** ± 4.96	**89.68** ± 4.73

TL: Transfer learning with ImageNet is used to initialise models' weights and then fine-tuned on Nerthus images.

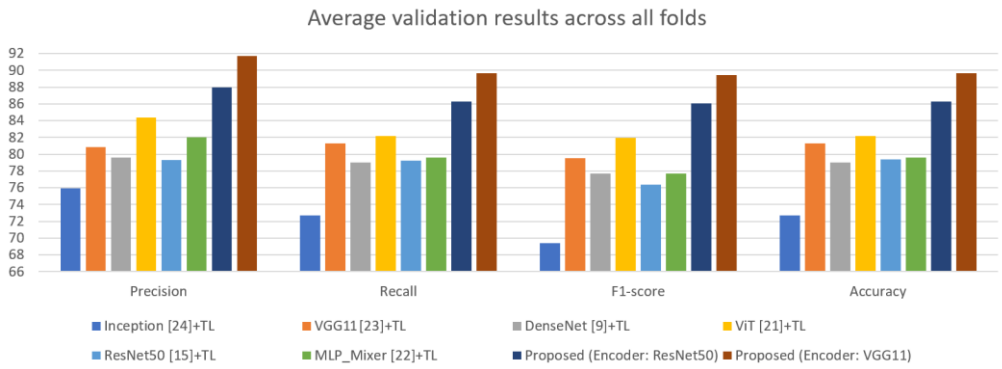


Figure 17 The average over the two-fold cross-validation is depicted. The proposed model achieved the best results across all metrics (i.e., precision, recall, F1-score, and accuracy).

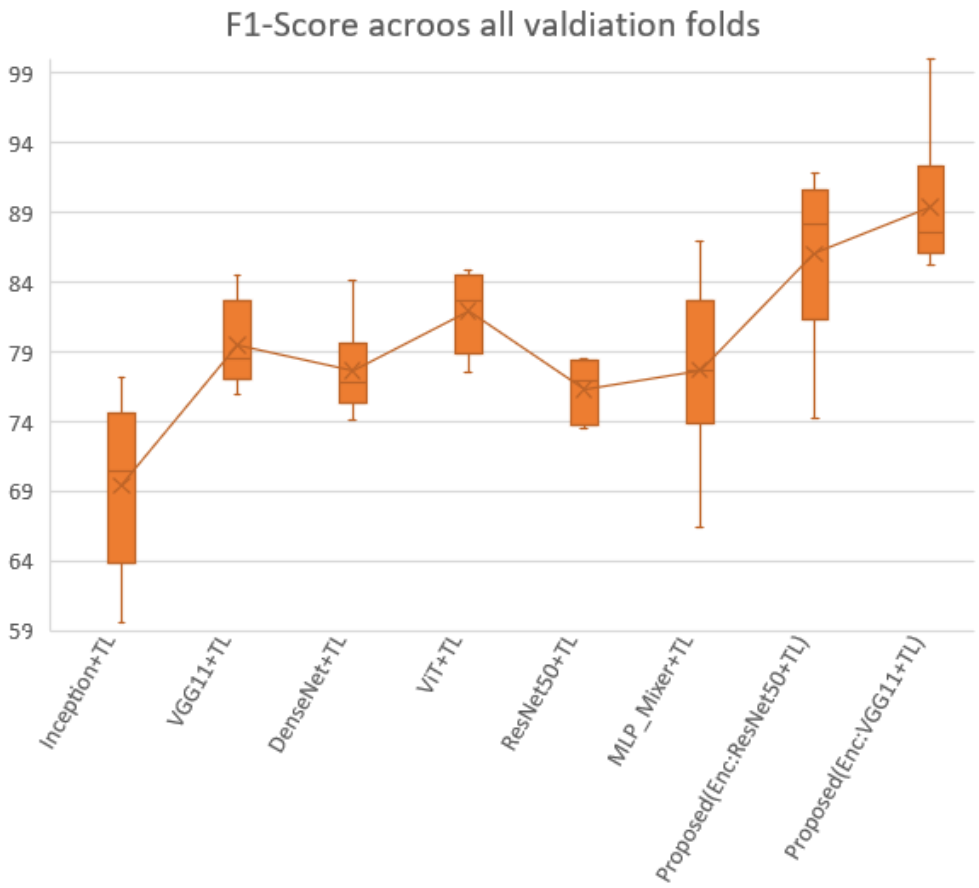


Figure 18 Whisker plot created based on the F1-score over all validation folds. The cross (x) inside the boxes represents the mean, meanwhile, the horizontal line (-) inside the box represents the median.

Considering only the F1-score, the proposed model achieved an approximate 10% improvement compared to the corresponding state-of-the-art models, namely VGG11 and ResNet50. These findings suggest that incorporating temporal information by considering sequential consecutive frames in a video enhances the overall performance, in contrast to treating each frame as an independent entity. To gain further insight into the results achieved by the proposed model, two different 2D projection mappings is used to visualise the feature vectors produced by the proposed Multiplexer and GRU, as seen in Figure 20. More comprehensive details regarding these results will be elaborated upon in section 3.3.5.

Proposed model against video-level models

To the best of our knowledge, no existing research paper has utilised the "Nerthus" dataset or any other private bowel preparation dataset as a collection of videos, focusing on video-level classification rather than individual images. Accordingly, in order to further validate the effectiveness of the proposed model, a comparison between our video-level model and other state-of-the-art video models from different domains are conducted. The performance results of the proposed model, along with the state-of-the-art video models, are presented in Table 8.

Table 8 The average and standard deviation are calculated over all validation folds (i.e., 6 folds in total). The first and second highest results are highlighted.

Models	Precision	Recall	F1-score	Accuracy
C2D [120]	63.18 ± 6.5	74.81 ± 5.98	67.86 ± 6.24	74.81 ± 5.98
I3D [121]	60.45 ± 4.28	66.76 ± 5.59	61.78 ± 5.07	66.76 ± 5.59
Slow [122]	67.04 ± 11.38	75.43 ± 9.76	69.89 ± 10.8	75.43 ± 9.76
C2D + TL	70.14 ± 10.79	74.86 ± 5.18	69.9 ± 7.03	74.86 ± 5.18
I3D + TL	87.28 ± 3.99	83.81 ± 6.98	82.1 ± 8.89	83.81 ± 6.98
Slow + TL	72.45 ± 10.34	72.11 ± 10.58	68.96 ± 10.38	72.11 ± 10.58
Proposed (Encoder: ResNet50)	**87.94** ± 4.83	**86.28** ± 6.28	**86.06** ± 5.92	**86.28** ± 6.28
Proposed (Encoder: VGG11)	**91.74** ± 3.95	**89.68** ± 4.73	**89.41** ± 4.96	**89.68** ± 4.73

TL: Transfer learning with Kinetics-400 is used to initialise video models' weights and then fine-tuned on Nerthus videos. Kinetics-400 contains 400 human action classes, with at least 400 video clips for each action.

The proposed model, irrespective of the chosen encoder, demonstrated state-of-the-art performance in terms of precision, recall, F1-score, and accuracy. This suggests that the architecture of the proposed model, which incorporates both temporal and key frames, outperforms models that solely rely on temporal frames. Notably, the utilisation of transfer learning, from the Kinetics-400 dataset, significantly improved the performance of video models, as observed in Table 8. Figure 19 further highlights that the use of transfer learning resulted in a better F1-score and precision for the used video models.

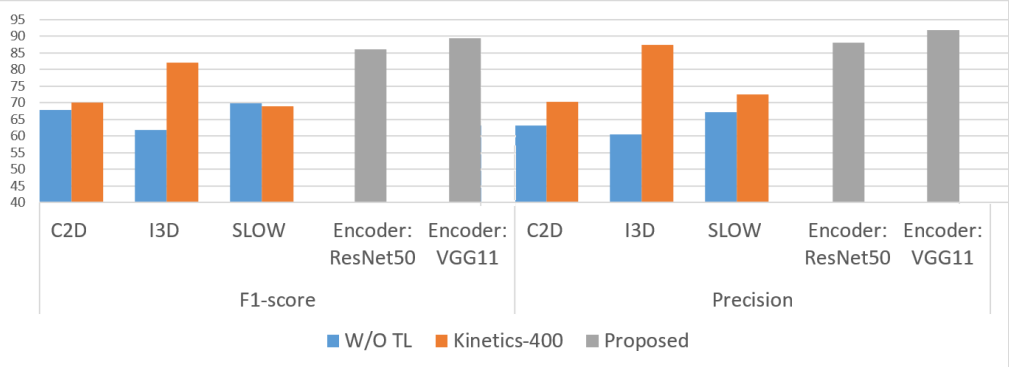


Figure 19 For each video-level model F1-score and Precision are depicted as bar chart.

Having established the superior performance of the proposed model compared to both frame-level and video-level models in the previous and current subsections, respectively, a comprehensive analysis of the proposed model from three different perspectives is elaborated in the upcoming sections. These include examining the impact of video sampling on the proposed model, visualising the feature vectors in a 2D space, and conducting a mathematical analysis of the prediction layer's gradients under two normalisation equations.

3.3.4 The effect of video size on the proposed model

In subsection 3.2.2, a method for representing videos as smaller representative samples (i.e., sub-video) was introduced. This is due to the limitation of the used GPU storage capacity. Each sub-video comprises 25 frames, which enabled creating training mini-batches with adequate size given the current limitation of the GPU. However, the impact of different sample sizes is also investigated in this thesis: 5, 15, 25, 35, and 45 frames per sub-video for "Dataset1", as illustrated in Table 9. Nevertheless, other variations are possible as well such as having 20-frames shift (i.e., 5, 25, and 45) or 5-frames shift (i.e., 5, 15, 20, 25, 30, 35, 40, and 45). However, both settings produce either less variations or too many variations which consume a lot of computational resources. Accordingly, a 10-frames shift is selected.

Table 9 The effect of sample size on the proposed model. The following results are an average of the 2-fold cross-validation of dataset1.

	Video sample's size (frame per sub-video)				
	5	**15**	**25**	**35**	**45**
Precision	84.6	89.86	89.97	90.14	93.66
Recall	84.22	89.87	87.32	87.4	91.79
F1-score	83.9	89.12	87.33	87.44	91.88
Accuracy	84.22	89.87	87.32	87.4	91.79

The results obtained for each sample size were averaged across the validation folds (a total of 2 folds), as presented in Table 9. The results clearly demonstrate that the model performs the worst when the sample size is 5 frames per sub-video. With only 5 frames per video, there is a higher likelihood of missing crucial frames that are essential for accurately determining the overall class label. Additionally, videos with predominantly non-informative frames are more likely to occur, which can lead the model astray during training. Consequently, the model's performance is negatively affected by such training batches.

On the contrary, when using 45 frames per sub-video, a more comprehensive representation is achieved, encompassing key frames, non-informative frames, and informative frames. Since the proposed model leverages both temporal frames and key frames, the presence of non-informative frames has a diminished impact on the model's performance. The same rationale can be applied to sub-video sizes of 15, 25, and 35 frames.

3.3.5 Visualising the produced feature vector

The aim of this section is to visualise and compare the feature maps generated by the "Multiplexer+GRU" layer against those produced by the encoder (specifically, ResNet50+TL). Furthermore, a thorough analysis of the fully-connected layer responsible for the classification decision is provided in section 3.3.6.

To gain insight into the impact of the proposed layers, a visualisation of the generated feature maps is presented in Figure 20. Specifically, feature maps representing frames of "Dataset2-fold1" validation set were selected, as outlined in Table 3. To avoid any biases towards favouring the proposed model, a deliberately chosen validation fold that doesn't represent the best-case scenario for the proposed model is selected for evaluation. Two visualisation techniques, namely t-distributed stochastic neighbour embedding (t-SNE) [112] and principal component analysis (PCA) [111], have been employed. It is important to note that the ResNet50 encoder [7] is designed to represent features for individual frames, while the proposed layer is trained to represent videos consisting of 25 frames, as discussed in section 3.2.2. Both t-SNE and PCA maps high-dimensional vector space to a lower-dimensional vector space, hence facilitating vectors' visualisation. In Figure 20, each point in the subfigures within the left column represents a feature map vector generated by the encoder layer of the proposed model, while each point in the subfigures within the right column represents a sub-video (i.e., 25 frames) generated by the "Multiplexer+GRU" layer.

In Figure 20, the t-SNE visualisation showcases the feature vectors generated by the encoder (ResNet50) as a series of interconnected strings. This implies that consecutive frames belonging to the same video exhibit similar features, resulting in similar embeddings. However, it is observed that some of the t-SNE encoder embeddings cluster in the middle, indicating that the encoder struggles to differentiate certain frames from different classes. This directly affected the classification results. On the other hand, the embeddings generated by the proposed "Multiplexer+GRU" layer are more scattered, indicating that videos from different classes are assigned to distinct feature vectors.

As shown in Figure 20, principal component analysis (PCA) embeddings for both the encoder and the proposed layer "Multiplexer+GRU" are shown in the second row. Unlike t-SNE, PCA captures the global structure by preserving the overall properties (i.e., eigenvectors correspond to high variance). The encoder embeddings exhibit similarities between neighbouring classes. For example, "class 2" (small fragment of stool on the mucosa) shares similar features with "class 1" (residual stool) and "class 3" (clear mucosa).

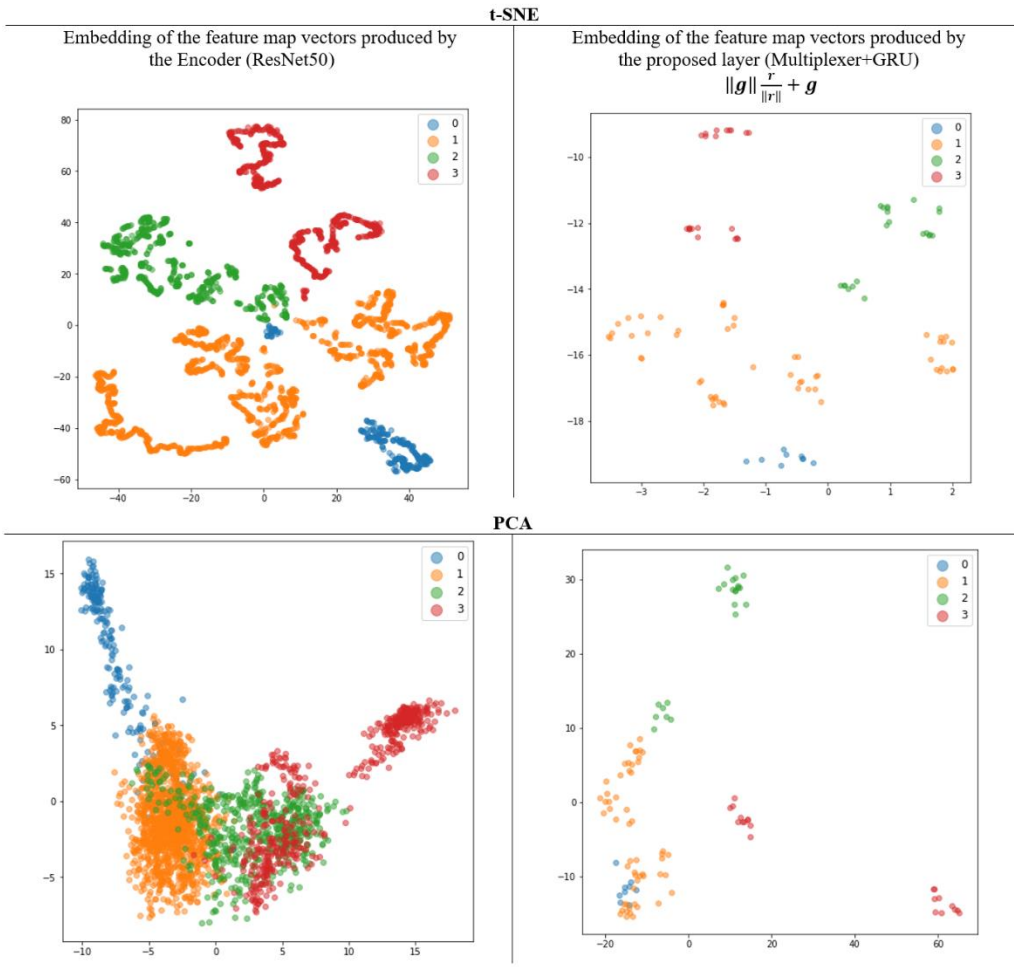


Figure 20 t-SNE and PCA embedding for the feature vectors produced by the encoder ResNet50 and the one produced by the proposed Multiplexer+GRU layer.

In contrast, the PCA embedding of the proposed layer "Multiplexer+GRU" shows a distinct separation between "class 1" and "class 3". However, there is some overlap between "class 0" and "class 1", indicating that the proposed layer didn't completely differentiate between them. Nevertheless, the proposed layer performs relatively better than the encoder, as evident from the embeddings in Figure 20 and the metrics' results in Table 7.

3.3.6 Gradient analysis of the proposed normalisation

This section focuses on investigating the impact of normalising the vectors $\boldsymbol{r}$ and $\boldsymbol{g}$ on the performance of the proposed model, as described in section 3.2.4 and outlined by Equation (9). The detailed calculation of the prediction layer's gradients can be found in Appendix C. However, in this section, analysis of the performance of the proposed model in relation to the chosen normalisation method is discussed, taking into account the computed gradients presented in Appendix C.

The vectors $\boldsymbol{r}, \boldsymbol{g} \in \mathbb{R}^{2048}$ are produced by GRU [109] and the proposed Multiplexer layer, respectively. The vector $\boldsymbol{g}$ is considered to be the main feature map vector since it represents the entire frames sequence, meanwhile, the vector $\boldsymbol{r}$ is utilised to support $\boldsymbol{g}$. To balance the effect of each vector, their

magnitude needs to be controlled by normalising their magnitudes. The magnitude of vector $\boldsymbol{r}$ can be normalised to be equivalent to the magnitude of the vector $\boldsymbol{g}$.

The vector $\boldsymbol{g}$ serves as the primary feature map vector as it represents the entire sequence of frames, while the vector $\boldsymbol{r}$ supports $\boldsymbol{g}$. In order to balance the impact of each vector, it is necessary to control their magnitudes through normalisation. One approach is to normalise the magnitude of vector $\boldsymbol{r}$ to be equivalent to that of vector $\boldsymbol{g}$:

$$\boldsymbol{q}_1 = \boldsymbol{g} + \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\|\boldsymbol{g}\| \tag{16}$$

Alternatively, both vectors can be normalised to be unit vectors, hence, both vectors would have the same magnitude:

$$\boldsymbol{q}_2 = \frac{\boldsymbol{g}}{\|\boldsymbol{g}\|} + \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|} \tag{17}$$

Please note that the normalisation applied in Equation (16) and Equation (17) ensures that the magnitude of vector $\boldsymbol{r}$ matches the magnitude of vector $\boldsymbol{g}$. However, it's important to consider that these normalisation equations will have an impact on the gradients of the fully-connected layer responsible for making the decision, as illustrated in Figure 21 and formulated as follows:

$$\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{q}) = \boldsymbol{\theta}\boldsymbol{q} + \boldsymbol{b} \tag{18}$$

$$\hat{\boldsymbol{y}} = SoftMax(\boldsymbol{z}) \tag{19}$$

$$SoftMax(z_i) = \frac{e^{z_i}}{\sum_{l=1}^{n} e^{z_l}}, \forall i = \{1, \dots, n\} \tag{20}$$

where the function $\boldsymbol{f}(\cdot)$ is an affine transformation with learnable weights $\boldsymbol{\theta} \in \mathbb{R}^{4\times 2048}$ and a learnable bias $\boldsymbol{b} \in \mathbb{R}^4$. The vectors $\boldsymbol{z}$ and $\hat{\boldsymbol{y}}$ are produced by the function $\boldsymbol{f}$ and a given discrete probability vector (i.e., target vector), respectively. The variable $n = 4$ since there is in total four classes in Nerthus dataset. The SoftMax function doesn't have any learnable parameters, hence, the layer that makes the classification decision is the fully-connected layer, as shown in Figure 21.

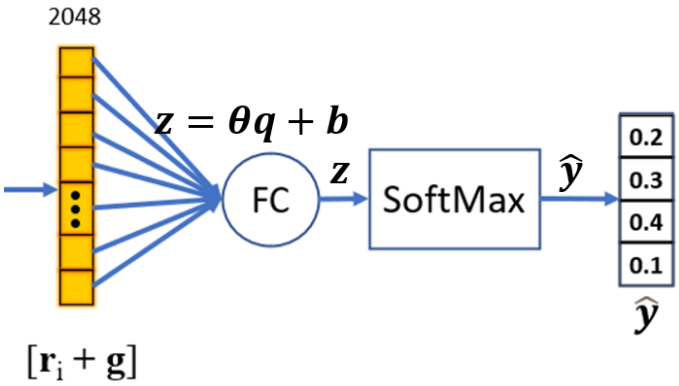


Figure 21 depicts the last fully connected layer that is responsible for generating probabilities vector.

The used loss function of the proposed model is negative log likelihood "$\mathcal{L}$":

$$\mathcal{L}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \qquad (21)$$

where the vector **y** represents the target discrete distribution and it is a one hot vector (e.g., $\boldsymbol{y} = [0010]^T$). The loss function $\mathcal{L}(\cdot)$ takes two vectors of size n and produces one real number:

$$\mathcal{L}: \mathbb{R}^n \rightarrow \mathbb{R} \qquad (22)$$

The gradients of the fully-connected layer in Figure 21 with respect to the loss function $\mathcal{L}$ is given by the following:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}} \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}} \qquad (23)$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{b}} = \frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}} \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{b}} \qquad (24)$$

The gradients of Equation (23) is derived in Appendix C:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = (\hat{\boldsymbol{y}} - \boldsymbol{y})^T \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}, \in \mathbb{R}^{1\times(4\times2084)} \qquad (25)$$

$$\frac{\partial z_i}{\partial \boldsymbol{\theta}} = \begin{bmatrix} \mathbf{0}^T \\ \vdots \\ \boldsymbol{q}^T \\ \vdots \\ \mathbf{0}^T \end{bmatrix} \in \mathbb{R}^{1\times(4\times2048)}, \forall i = \{1, \dots, n\} \qquad (26)$$

where $n = 4$ due to having four classes (i.e., bowl cleansing degree). Since the gradients of the loss function $\mathcal{L}$ w.r.t weights $\boldsymbol{\theta}$ is calculated, the effects of the applied normalisation for Equation (16) and Equation (17) can be analysed by calculating the magnitude of the gradients $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}$. The lower and upper bound of the Frobenius norm $\|\cdot\|_F$ of the gradients $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}$ can be calculated, given that $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ each represents a discrete probability vector:

$$\left\|\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}\right\|_F = \left\|(\hat{\boldsymbol{y}} - \boldsymbol{y})^T \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}\right\|_F \leq \|(\hat{\boldsymbol{y}} - \boldsymbol{y})^T\|_F \cdot \left\|\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}\right\|_F \quad (27)$$

$$\left\|\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}\right\|_F \leq c \cdot \left\|\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}\right\|_F \quad (28)$$

where c is just a constant term. The Frobenius norm in Equation (28) can be calculated given the gradients of $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}$ in Equation (26):

$$c \cdot \left\|\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}\right\|_F = c \cdot \|4\boldsymbol{q}\| = c \cdot \|\boldsymbol{q}\| \quad (29)$$

$$\therefore \left\|\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}\right\|_F \leq c \cdot \|\boldsymbol{q}\| \quad (30)$$

Since the gradients $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}$ contains only zeros and four **q**'s, the Frobenius norm $\left\|\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}\right\|_F$ would be equal to the Euclidean norm of $\|4\boldsymbol{q}\|$. It is noted that the lower and upper bounds of the gradients' magnitude $\left\|\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}\right\|_F$ are 0 and $c \cdot \|\boldsymbol{q}\|$, respectively. Given Equation (30), the effect of normalising the vectors **r** and **g** in both Equation (16) and Equation (17) can be clearly contrasted. For Equation (16) the upper bound of the gradients' magnitude is:

$$c \cdot \|\boldsymbol{q}_1\| = c \cdot \left\|\boldsymbol{g} + \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\|\boldsymbol{g}\|\right\| = c \cdot \|\boldsymbol{g}\| \left\|\frac{\boldsymbol{g}}{\|\boldsymbol{g}\|} + \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\right\| \quad (31)$$

$$\leq c \cdot \|\boldsymbol{g}\| \left(\left\|\frac{\boldsymbol{g}}{\|\boldsymbol{g}\|}\right\| + \left\|\frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\right\|\right) = c \cdot \|\boldsymbol{g}\| \quad (32)$$

Meanwhile for Equation (17), the upper bound of the gradients' magnitude is:

$$c \cdot \|\boldsymbol{q}_2\| = c \cdot \left\|\frac{\boldsymbol{g}}{\|\boldsymbol{g}\|} + \frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\right\| \leq c \cdot \left(\left\|\frac{\boldsymbol{g}}{\|\boldsymbol{g}\|}\right\| + \left\|\frac{\boldsymbol{r}}{\|\boldsymbol{r}\|}\right\|\right) = \mathrm{c} \quad (33)$$

In conclusion, the range of the gradients' magnitude $\left\|\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}\right\|_F$ is approximately given by the interval $(0\ ,\ c \cdot \|\boldsymbol{g}\|)$ and $(0\ ,\ c)$ when normalising the feature vectors using Equation (16) and Equation (17), respectively. It is noticed that gradients' magnitude $\left\|\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}\right\|_F$ will be bounded by a constant range if the feature vectors $\boldsymbol{r}$ and $\boldsymbol{g}$ are normalised to a unit vector as in Equation (17). On the other hand, normalising vector $\boldsymbol{r}$ to have a similar magnitude as vector $\boldsymbol{g}$ will result in having a various/stochastic gradients' magnitudes. This eventually induces elements of stochasticity regards updating the parameters.

The models' parameters were updated using stochastic gradient descent with momentum, which led to reaching and escaping several local minima. This information is supported by Figure 22-(a), which demonstrates the optimization process and the behaviour of the loss function during training.

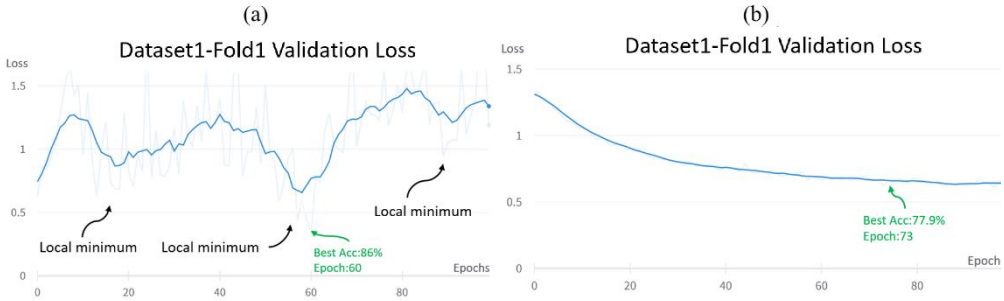


Figure 22 Dataset1-Fold1 validation loss of the proposed model when (a) normalising the vector **r** to have similar magnitude as **g** and (b) normalising both vectors **r** and **g** to become a unit vector.

In Figure 22, the comparison of validation losses between Equation (16) and Equation (17) is shown in columns (a) and (b), respectively. The validation set used for this comparison is Dataset1-Fold1, which can be referred to in Table 2. To highlight the pattern of the validation loss, a moving average with a window size of 10 is applied. The window size was determined via trial and error.

The epoch that achieved the best validation accuracy is highlighted in green. It is noticed that there is a significant improvement in accuracy, approximately 13%, when using Equation (16) compared to Equation (17). This improvement can be attributed to the model's ability to escape multiple local minima, as depicted in Figure 22-(a).

It is worth noting that the epoch with the lowest validation loss does not necessarily align with the epoch that achieves the best validation accuracy. However, in this specific experiment, it happened that the epoch with the lowest loss in Figure 22-(a) also corresponded to the best validation accuracy, which was approximately 86%.

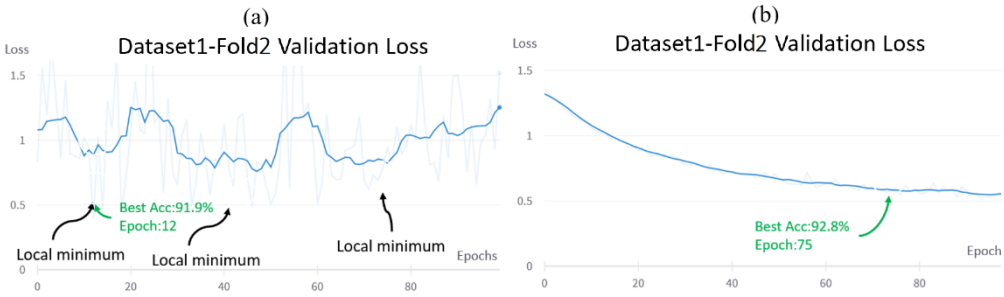


Figure 23 Dataset1-Fold2 validation loss of the proposed model when (a) normalising the vector **r** to have similar magnitude as **g** and (b) normalising both vectors **r** and **g** to become a unit vector.

In the case of the dataset Dataset1-Fold2, it is observed from Figure 23 that the proposed model achieved high validation accuracy regardless of the normalisation method used. Specifically, when Equation (16) was used for normalisation, the validation accuracy was 91.9%, and when Equation (17) was used, the accuracy increased to 92.8%. This indicates that the proposed model might have found a good local minimum when both vectors (r and g) were normalised to unit vectors.

However, the difference in performance between the two normalisation methods, Equation (16) and Equation (17), is only 0.9% for the dataset Dataset1-Fold2. In this case, using Equation (16) to escape local minima did not yield better results than using Equation (17), as depicted in Figure 23.

Nevertheless, the difference in performance between the two normalisation methods is insignificant compared to the significant improvement achieved using the dataset Dataset1-Fold1, which corresponds to approximately 13%. Therefore, it is decided to choose Equation (16) as the normalisation method, despite the slight sacrifice in performance in some validation folds.

3.4 Summary

In this chapter, the performance of various deep learning methods for assessing bowel clarity is investigated by leveraging a public database named Nerthus. It is shown in this chapter that the literature achieved over-estimated performance due to improper treatment of this video dataset. Dealing with the dataset as a collection of independent frames/images, and randomly dividing the images into training and validation, resulted in creating nearly identical distributions (training and validation) and thereby achieving amplified performance. Furthermore, it is advisable to consider videos when testing a model to mimic real-clinical environment. Accordingly, it is proposed to treat Nerthus dataset as a collection videos, instead of independent frames/images, to create various validation folds. Given the limited capacity of the used GPU, a sampling method is proposed to create smaller representative sub-videos to enable training the proposed model with adequate batch size.

The proposed model is designed to leverage temporal information within videos, as well as spatial information within individual frames. The proposed model consists of mainly four components, namely, an encoder that encode input frames into feature-vectors, a layer that handle sequenced data (i.e. GRU), a layer that select a representative feature-vector (i.e. Multiplexer), and finally a fully-connected layer that generates probability distribution for the target labels. The produced vectors by the Multiplexer and GRU layers are normalised and added together to increase the inter-class separability and intra-class compactness. This effect has been visualised in a 2D plot using the principal component analysis (PCA) as well as t-distributed stochastic neighbour embedding (t-SNE).

The extensive experiments on different validation folds indicate that the proposed model achieved better than the state-of-the-art deep learning models. The proposed model achieved the highest record on Precision, Recall, F1-score, and Accuracy with approximately 10% enhancement, on average, compared with the best performing frame-level state-of-the-art models.

Moreover, the proposed model outperformed video-level state-of-the-art models across all metrics. Specifically, when considering only the F1-score, the difference between the best video-level model and the proposed model amounts to 7.31%. The performance improvements across different metrics indicate the effectiveness and superiority of the proposed approach in comparison to other models.

It is hoped that this work would spark interest in the research community to further investigate this problem and construct public annotated datasets that enable appropriate evaluation of the generalisability of models.

Chapter 4

Polyp Segmentation

4.1 Introduction

Polyps are abnormal growth in the mucosa which signifies a risk of developing cancer if left untreated, as seen in Figure 24. Hence, it is essential to identify and analyse any formed polyp by colonoscopist as soon as it appears. Colorectal cancer is a prevalent type of cancer in which it is considered the third leading cause of cancer deaths [74]. Polyp recognition and treatment often pose significant challenges due to the complexity of the anatomical structure of the colon and rectum, making such screening require a high level of expertise. Furthermore, the irregularity of polyp shapes and human-based errors such as being prone to fatigue could affect the quality of screening. Furthermore, colonoscopists' expertise level effect polyp missing rate. In fact non-gastroenterologists are five times more likely to miss colorectal cancer during colonoscopy than gastroenterologists [123]. To address these confronts, computer systems have been proposed as a promising assistance. These systems aim to reduce human subjectivity and enhance polyp detection rates.

Recently, computer vision community has signified a considerable interest in deep learning models due to their ability to exceed hand-crafted classifiers regularly portrayed as conventional machine learning models [124]. Nevertheless, deep learning models suffer from a pivotal set of challenges. Challenges such as the need for solid training datasets, addressing the issue of overfitting, and effectively tuning hyperparameters to achieve optimal performance. As a result, a notable decline in performance is observed when these models are tested on unseen future samples [32]. Accordingly, several architecture designs are presented in the literature [99], [124]–[129]. Nevertheless, these models, by default, inherit the limits of deep learning methods in which the accessibility of adequate dataset is vital. Currently, this is not the case in the colonoscopy domain which automatically leads to the usage of image augmentation. However, such simple augmentation methods such as rotation, flipping, and shearing would not significantly enhance the performance on unseen data due to some invariants properties entrenched in deep learning models [130], [131].

Utilising generative adversarial networks (GANs) are considered admirable alternatives to expand the training dataset. Nevertheless, generating synthetic polyp images using GANs is not a clear-cut procedure, and, in some incidence, it needs manual intervention [14], [15]. In [15], an edge filtering-based conditioned image mask is proposed to train conditional GAN [132]. Meanwhile, [14] used real non-polyp images and converted them to polyp

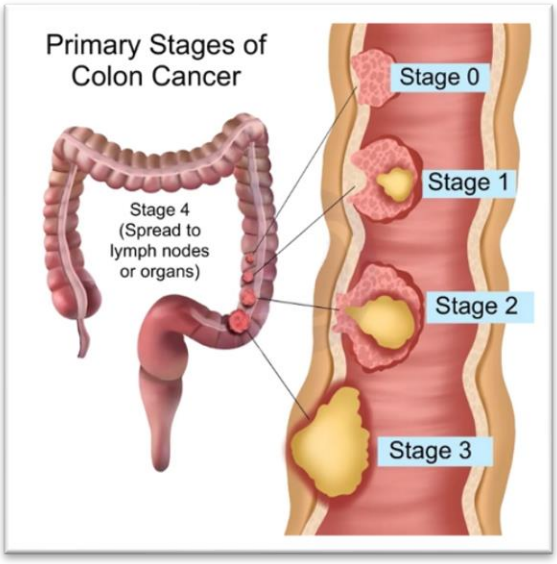


Figure 24 If polyp left untreated it may develop to a cancer. Image courtesy of [133].

images using GAN inpainting model. The earlier method requires pre-processing time to merge the extracted edge-filtering images and random polyp masks, whereas the latter requires post-processing to verify that synthetic polyps are realistic [134].

To address these limitations, a synthetic data generation pipeline called SinGAN-Seg is proposed by [134]. As opposed to [14], [15], SinGAN-Seg generate synthetic images along with corresponding labels without the need for doing post-processing or manual work. Essentially, they utilised SinGAN [135] to generate images with masks and utilised a style-transfer method [136] to fine-tune them. Although their proposed method produced realistic polyp images, the inflated dataset did not enhance the performance of the used segmentation model except when the training data was extremely small (i.e., less than 20 images).

It is noticed in general that generating synthetic images using GANs is a time-consuming procedure, demands high computational complexity, and unstable. Furthermore, the anticipated segmentation enhancement is not promising. In fact, a comprehensive study was conducted by [137] to compare the effectiveness of conventional augmentation methods and GANs encompassing sophisticated techniques. They concluded that traditional augmentation techniques remain the most fruitful [12], [137]. The reason is that GANs based models sample its data from the training distribution which yield images that have similar features as the original training images which turns to be not beneficial in enhancing generalisability of deep learning models.

This thesis presents an alternative approach to improve generalisability without explicitly increasing the dataset size. Instead of augmenting or adding new images to the dataset, a segmentation model is exposed to out-of-domain images during training epochs. The method involves gradually transforming the original images by manipulating their texture meanwhile feeding those on-the-

fly transformed images to the segmentation model to acquire it invariant properties towards colour and texture changes. Two concrete implementations stemmed out of this proposed framework, including, minimizing images' total variations $TV_{\emptyset}$ and texture interpolations $TI_{\emptyset}$. The $TV_{\emptyset}$ objective is to gradually wash-out background's textures (i.e., the gradient of the background ≈ 0) while maintaining the region of interest's texture (i.e., polyp). On the other hand, $TI_{\emptyset}$ applies spatial interpolation between an input image with fine-grained texture and a corresponding version with reduced texture details. By training on a range of texture and colour variations, the segmentation model becomes more robust and adaptable to different conditions. Each one of the proposed models is elaborated independently in the Methodology section. Both models were tested against the state-of-the-art models and showed superior generalisability results on unseen test sets from different medical centres.

4.2 Methodology

In the first section, a proposed framework will be presented from a general perspective. Following that, two different concrete implementations of the proposed framework will be elaborated in two subsequent sections. Finally, the used datasets for experiments will be demonstrated.

4.2.1 Proposed framework: An overview

Polyps do not exhibit specific shapes, colours, or sizes, and the elasticity of the colon lining (i.e., mucosa) can sometimes look like polyps. These factors significantly affect the performance of deep learning models used for polyp segmentation. Moreover, worse generalisability performance is anticipated when these models are tested on unseen polyp images from different medical centres. Accordingly, their suitability for clinical practice is questionable. Due to the scarcity of proper datasets, polyp segmentation remains a challenging problem.

To address this issue, researchers often opted to techniques such as data augmentation or generative adversarial networks (GANs) to artificially increase the size of available datasets. However, these approaches have not been proven to significantly enhance generalisability performance. In contrast, this section proposes a novel approach that involves on-the-fly image-to-image transformation during the training phase. The goal is to introduce out-of-domain samples in each epoch to regularize a segmentation model $f(\cdot)$, as illustrated in Figure 25.

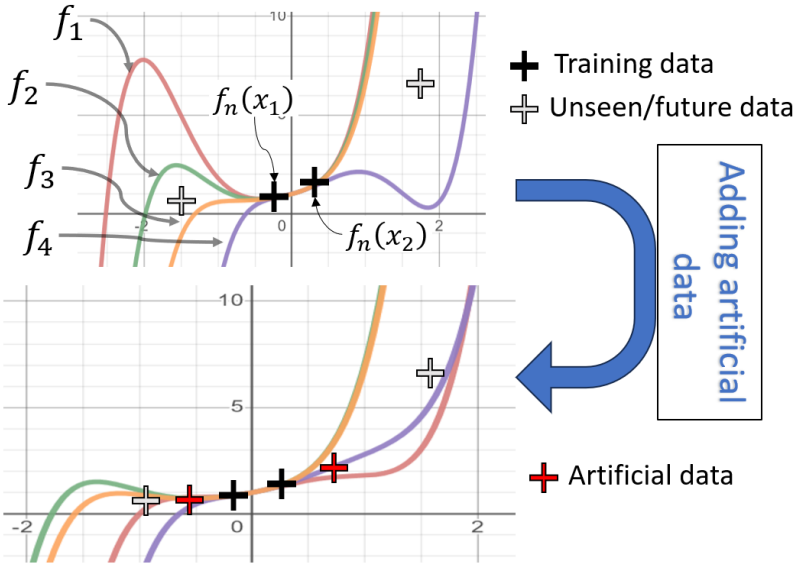


Figure 25 The effects of introducing artificial samples on the mapping functions (i.e., learnable models).

In essence, deep learning model is a function governed by a set of parameters $f(x;\theta)$. The function maps inputs to outputs $f_\theta: X \to \tilde{Y}$ and both the inputs and the outputs could be any mathematical object, including, number, vector, matrix or an image. Furthermore, a loss function is utilised to quantify mapping errors $L: \tilde{Y} \to \mathbb{R}$ and accordingly tune function's parameters θ to get the desired output close to ground truth Y. Such a function can be called a "model".

Given limiting data X and the complexity of the model f_θ, the latter can overfit the training data in various ways, as seen in Figure 25. Furthermore, there is no unique solution to the mapping problem, hence, several different set of parameters θ can drive the model to have the desired output. However, the model f_θ may not perform well on future unseen data due to lack enough training samples (i.e., poor generalisability).

To address the poor generalisability problem, researchers tend to regularize their model using L1 & L2 regularization, dropout, early stopping, data augmentation, and inflating dataset using Generative Adversarial Networks (GANs). It is true that these methods reduce overfitting problems, however, it may not enhance the generalisability of a model if the distribution of future data is shifted from the distribution of training data. In other words, if the model has not been exposed to patterns similar to future data, reducing overfitting will not improve its generalisability. Moreover, inflating the training set with samples taken from a distribution that resembles the training distribution will not enhance the generalisability. To that end, a deep learning framework that continuously exposes the model to out-of-domain samples is proposed, thus, enhancing its generalisability competence, as seen in Figure 26.

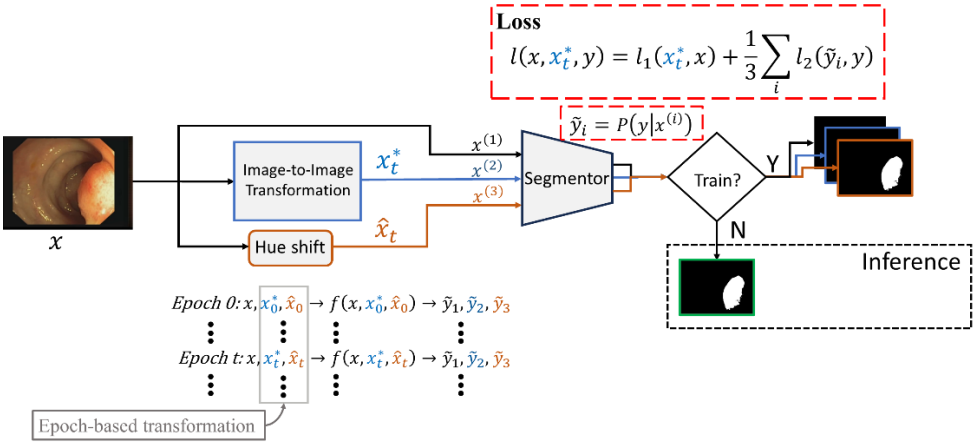


Figure 26 Overview of the proposed framework. The input image x is independently transformed by both a random hue-shift function and image-to-image transformation unit.

The proposed framework consists of mainly three components, including an image-to-image transformation unit $T_\emptyset$, a random hue shift unit H, and a segmentation model f_θ parameterized by θ. For each epoch t, the learnable unit $T_\emptyset$ responsible for continuously transforming input images' texture $T_\emptyset(x,t) = x_t^*$, meanwhile, the unit H randomly shift the colour of input images $H(x,t) = \hat{x}_t$. Here, the goal is not inflating the number of training samples but rather apply online transformations to input images during training epochs. More details about these transformations are further elaborated in the subsequent sections.

A colourful image $x \in \mathbb{R}^{3\times H\times W}$ can be seen as a point in high-dimensional space $\mathbb{R}^k$, as depicted in Figure 27. According to manifold hypothesis, training images cluster in a small subset of the high-dimensional space $\mathbb{R}^k$ [114], where $k < 3 \cdot H \cdot W$. This subspace is referred to as training manifold. Outside the training manifold resides noisy images as depicted in Figure 27. Given this manifestation, the essence of the proposed framework is to produce samples nearby the training manifold by leveraging image-to-image transformations. By introducing variations to the original input, the used model would be introduced to new patterns and get regularized. As a result, the generalisability of the used model would be enhanced.

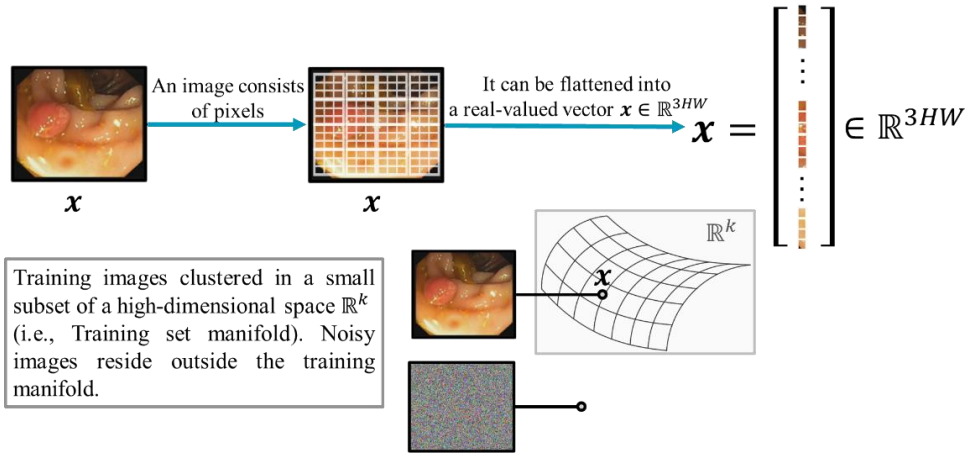


Figure 27 An image can be seen as a point in high-dimensional space. Each pixel is considered a single dimension.

The overall training objective to be minimized by the proposed framework is illustrated in Figure 28. The segmentation model f_θ calculates the likelihood of having polyp regions given an input image x (i.e., $P_\theta(y|x)$). Accordingly, the segmentation model receives images x as input and produce $\tilde{y}$ as an output $\boldsymbol{f_\theta}: \boldsymbol{X} \rightarrow \widetilde{\boldsymbol{Y}}$. The variable $\tilde{y}$ resembles an estimation to a target conditional likelihood distribution $y \equiv P(y|x)$, hence, $\tilde{y} \equiv P_\theta(y|x)$. Furthermore, log loss function is utilised to quantify the distance between the target distribution $P(y|x)$ and the estimate distribution $P_\theta(y|x)$. Accordingly, the segmentation model f_θ tries to minimize negative log-likelihood $-logP_\theta(y|x)$ in order to find parameters θ that best produce the target distribution $P(y|x)$ (i.e., polyp segmentation mask).

As shown in Figure 28, there are a set of images $\boldsymbol{X}$ and corresponding ground truth labels (i.e., masks) $\boldsymbol{Y}$ sampled from two distributions $\boldsymbol{X} \sim P_{data(x)}$ and $\boldsymbol{Y} \sim P_{data(y)}$, respectively. Added to that, for each training sample x, there are two conditional probability distributions, including $\widehat{\boldsymbol{X}} \sim H(\hat{x}|x)$ and $\boldsymbol{X}^* \sim T_\emptyset(x^*|x)$. As discussed above, H and $T_\emptyset$ are two different transformation units (more details in subsequent sections). If only the original training set $\boldsymbol{X}$ are considered for training, then the objective of the segmentation model is to minimize on average the negative log-likelihood $\mathbb{E}_{P_{data(x)}} - logP_\theta(y|x)$. On the other hand, if a set of transformed images $\widehat{\boldsymbol{X}}_{\boldsymbol{n}} \sim H(\hat{x}_n|x_n)$ that correspond to a single image x_n is considered for training, then the objective is to minimize on average the negative log-likelihood $\mathbb{E}_{H(\hat{x}_n|x_n)} - logP_\theta(y_n|\hat{x}_n)$. Likewise, if another set of transformed images $\boldsymbol{X}_n^* \sim T_\emptyset(x_n^*|x)$ is considered for training, then the training objective is to minimize $\mathbb{E}_{T_\emptyset(x_n^*|x_n)} - logP_\theta(y_n|x_n^*)$). However, if all the training images x are considered along with all applied transformations by H and $T_\emptyset$, then the training objective is to minimize the following nested negative-log likelihood:

$$-\mathbb{E}_{P_{data(x)}}\left[logP_\theta(y|x) + \mathbb{E}_{H(\hat{x}|x)}logP_\theta(y|\hat{x}) + \mathbb{E}_{T_\emptyset(x^*|x)}logP_\theta(y|x^*)\right] \quad (34)$$

Therefore, the training process of a segmentation model can be explained as performing stochastic gradient descent on the expectation in Equation (34). It is empirically demonstrated in the experiment section (i.e., Ablation study) that the best performance for the used segmentation model is achieved only if all terms in Equation (34) were considered for the training process. Deleting any term in Equation (34) negatively affects the performance of the segmentation model. Given the above presentation of the proposed framework, two concrete implementations are discussed in the following two subsections.

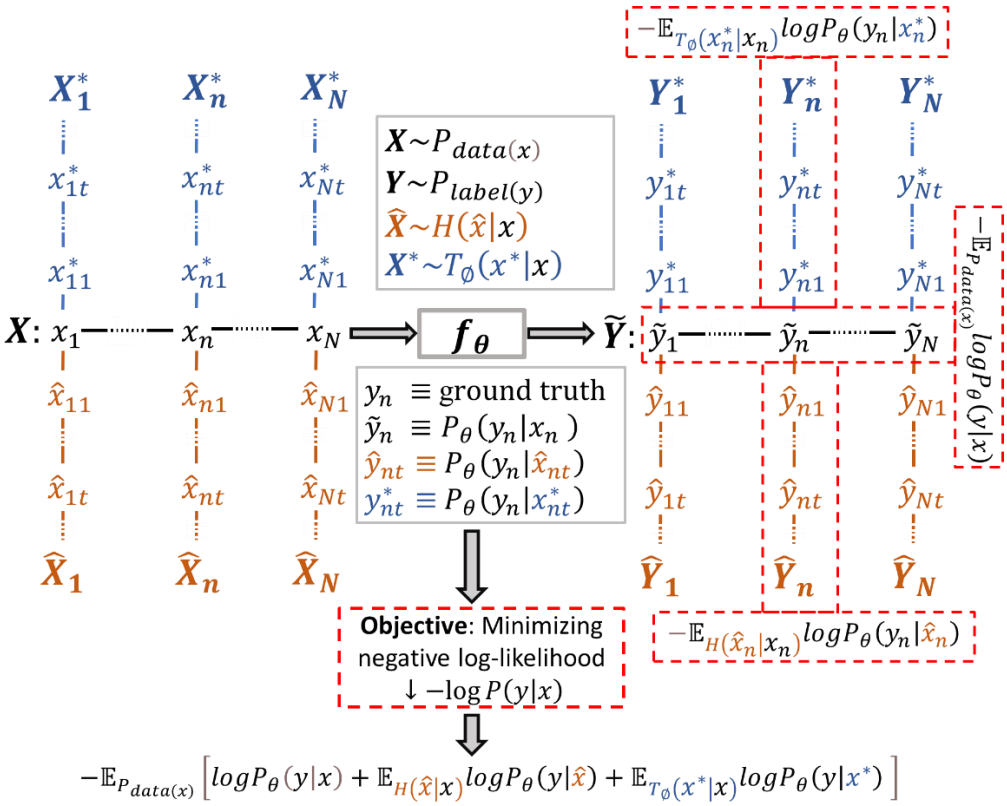


Figure 28 Proposed framework objective. For each training image x there are two series of transformations produced by H and T units, respectively.

4.2.2 Proposed model I: Total Variational

The proposed framework consists of mainly three components including $TV_\emptyset$ model that apply image-to-image transformation, a random colour shift unit, and a segmentation model as shown in Figure 29. For each epoch, different transformation is applied to an input image x.

To enhance the generalisability and robustness of any segmentation model, it should be exposed to images sampled from a probability distribution different than the training distribution $P_{data(x)}$. The intention of introducing out-of-domain samples is to regularize the used segmentation model, hence, preventing overfitting. Furthermore, introducing new patterns to the segmentation model will enhance its recognition competence. Accordingly, a novel image-to-image transformation is proposed that is inspired by Total Variational method [138].

Total Variational minimization was originally proposed to lessen noises of a given image q by introducing a denoised version g. The new denoised image g should have fundamental features of image q with minimum variations. This objective is formulated as follows:

$$\min_g \frac{\lambda}{2} \|q - g\|^2 + \|\nabla g\| \tag{35}$$

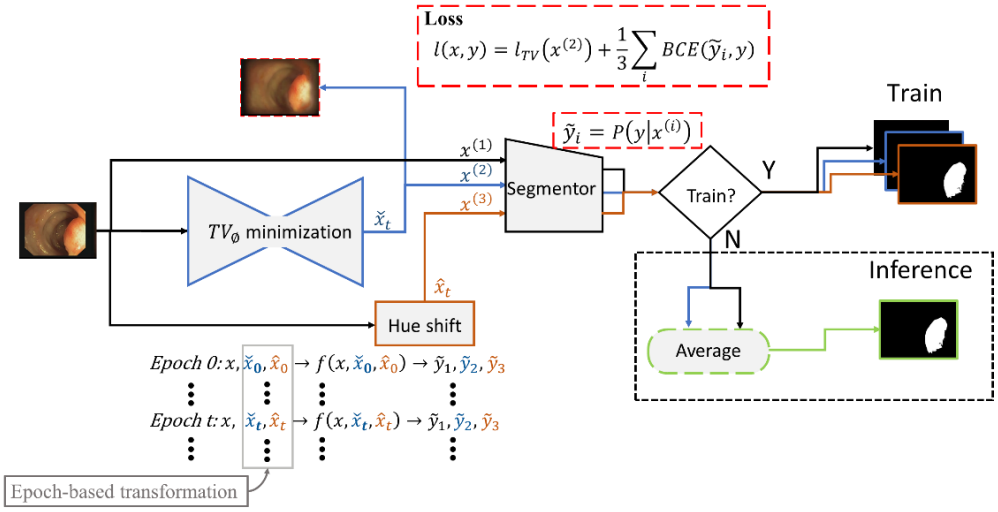


Figure 29 This figure depicts the first concrete implementation of the proposed framework. The input image is independently transformed by a random Hue-shift function and Total Variation minimization model TV_ϕ.

The first term measures the discrepancy between the original image q and the produced one g, meanwhile, the second term quantify the total variations $\|\nabla g\|$ of the generated image. The above equation is regularized by a scalar λ to balance between reconstruction term $\|q - g\|^2$ and denoising term $\|\nabla g\|$.

The concept of Total Variational minimization objective is leveraged by considering the entire background region as being noises that need to be eliminated. On the other hand, only the structure of the region of interest (i.e., polyps) is retained. This objective can be formulated as follows.

Let x and $\check{x}$ be two 2D images which represent original and transformed images, respectively. Furthermore, let $\boldsymbol{A}$ be set of points that represents polyp pixels' location. Then our proposed Total Variation minimization objective, which will be addressed by $TV_\emptyset$ deep learning model, is defined by:

$$\min_{\check{x}} \begin{cases} \alpha\|x(w,h) - \check{x}(w,h)\|^2 & for \quad (w,h) \in \boldsymbol{A} \\ \beta\|\nabla\check{x}\| & otherwise \end{cases} \tag{36}$$

where (w, h) represents pixel coordinates, whereby, α and β are scalars used to balance between construction and total variation $\|\nabla\check{x}\|$ objectives. Since the training images consists of RGB channels, the gradients for a single channel $\nabla\check{x}_c$ is defined as follows:

$$\nabla\check{x}_c = \left[\frac{\partial\check{x}_c}{\partial w} \quad \frac{\partial\check{x}_c}{\partial h}\right] \tag{37}$$

where c represent a RGB channel of a colourful image, meanwhile, w and h are the x-axis and y-axis coordinates of polyp pixel. Since the transformed image $\check{x}$ consists of 3 channels (i.e., Red, Green, and Blue), the total variations for colour images are defined as follows:

$$\|\nabla\check{x}\| = \sqrt{\|\nabla\check{x}_R\|^2 + \|\nabla\check{x}_G\|^2 + \|\nabla\check{x}_B\|^2} \tag{38}$$

where $\|\cdot\|$ is Frobenius norm, meanwhile, $\check{x}_R$, $\check{x}_G$, and $\check{x}_B$ are 2D Red, Green, Blue channels, respectively. To calculate the patrial derivatives $\frac{\partial \check{x}_c}{\partial w}$ and $\frac{\partial \check{x}_c}{\partial h}$ two 3-by-3 kernels are employed to the input image $\check{x}_c$ by convolution:

$$\frac{\partial \check{x}_c}{\partial w}(a,b) = k_w * \check{x}_c(a,b) = \sum_{i=-3}^{3}\sum_{j=-3}^{3} k_w(i,j)\,\check{x}_c(a-i,b-j) \quad (39)$$

$$\frac{\partial \check{x}_c}{\partial h}(a,b) = k_h * \check{x}_c(a,b) = \sum_{i=-3}^{3}\sum_{j=-3}^{3} k_h(i,j)\,\check{x}_c(a-i,b-j) \quad (40)$$

where k_w and k_h are Sobel operators:

$$k_w = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \; and \; k_h = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (41)$$

Other than Sobel operators could be used, and it would have some effect on produced images by the $TV_{\emptyset}$ model. The effect of using different derivative operators on the produced images is discussed in .

The $TV_{\emptyset}(\check{x}|x)$ is a deep learning model consists of Unet [8] followed by a sigmoid layer. The sigmoid layer is used to normalise the Unet output range to be $(0,1)$. The $TV_{\emptyset}$ has learnable parameters $\emptyset$ that get updated during the training epochs through stochastic gradient descent. Accordingly, $TV_{\emptyset}$ learns to identify polyp texture to preserve it, meanwhile smoothing the background in a progressive matter as seen in Figure 30 and Figure 31. More examples are presented in Appendix D.2.

The proposed model is designed to be an end-to-end deep learning model in which only one training phase are required to train both the TV and segmentation models as depicted in Figure 29. At epoch t, an original image x, a corresponding Hue shifted image $\hat{x}_t$, and a transformed image $\check{x}_t$ are produced by $TV_{\emptyset}$ model, all of which are fed to a segmentation model. Accordingly, the training process of a segmentation model can be viewed as performing stochastic gradient descent on the following expectation:

$$-\mathbb{E}_{P_{data(x)}}\left[logP_\theta(y|x) + \mathbb{E}_{H(\hat{x}|x)} logP_\theta(y|\hat{x}) + \mathbb{E}_{TV_{\emptyset}(\check{x}|x)} logP_\theta(y|\check{x})\right] \quad (42)$$

where $P_{data(x)}$ is the training distribution, $H(\hat{x}|x)$ represents a conditional distribution over Hue shifted samples $\hat{x}$, given a data sample x, and finally, $TV_{\emptyset}(\check{x}|x)$ represents a conditional distribution over transformed samples $\check{x}$, given a data sample x.

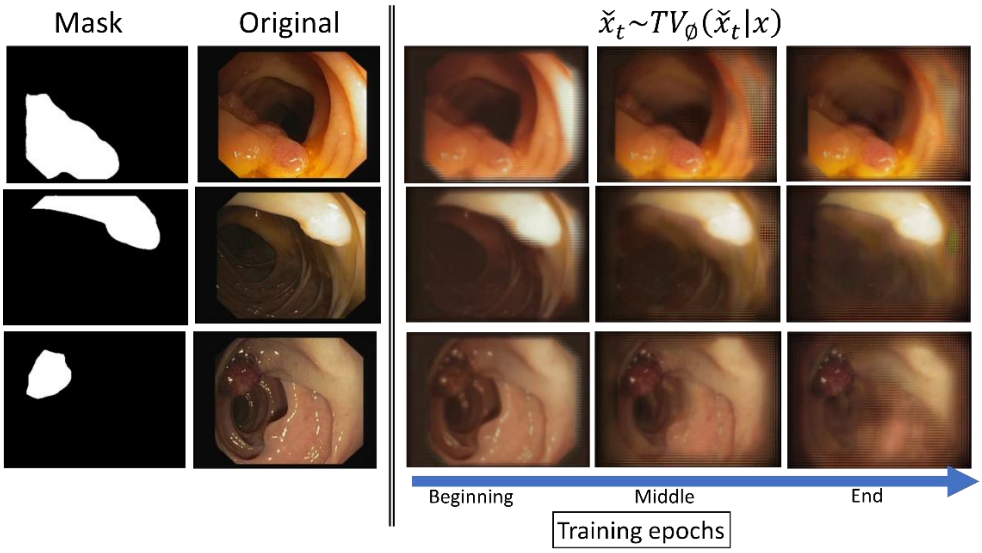


Figure 30 Illustrations of the transformation of input images using $TV_{\emptyset}$ model during training. While training is progressing, background textures are gradually washed out meanwhile polyp are retained.

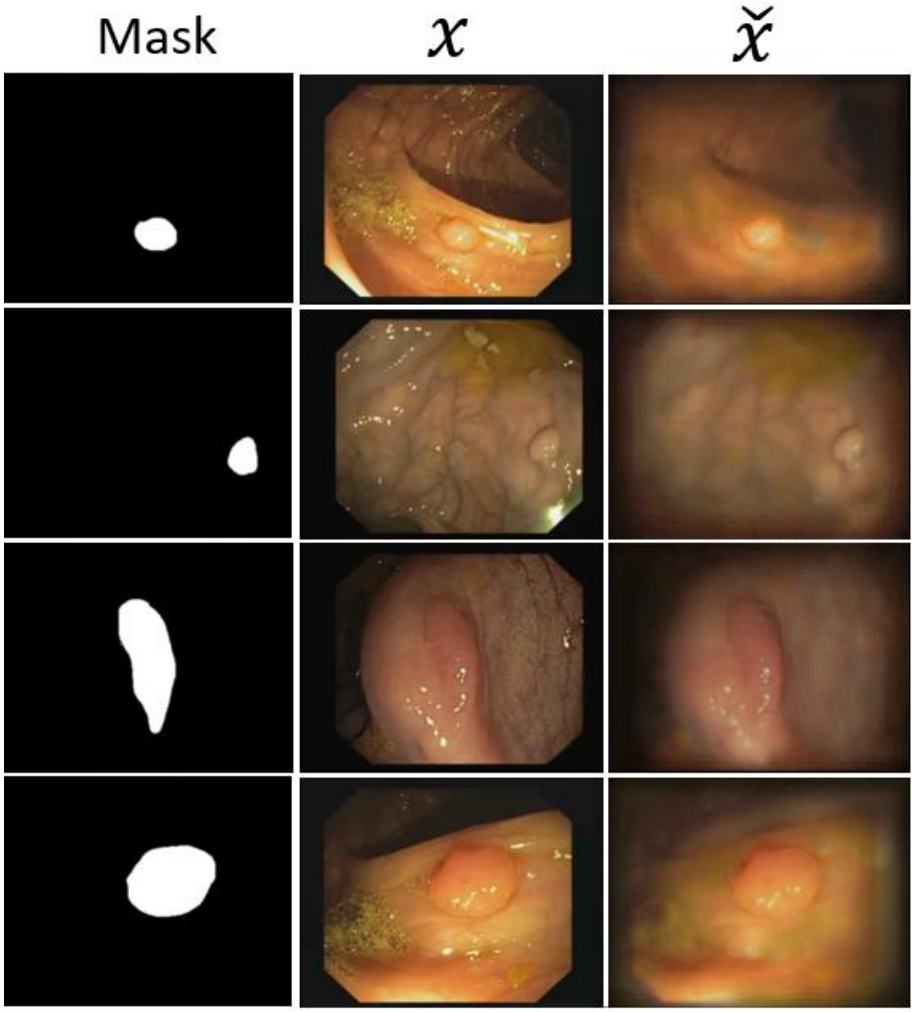


Figure 31 Examples of the transformed validation images using the proposed $TV_{\emptyset}$ model.

It is worth noting that it is the case that the total variations of a transformed image is always less than the original image (i.e., $\|\nabla \breve{x}_t\| < \|\nabla x\|$ for every epoch t). A segmentation model learns from three different samples (i.e., $x, \hat{x}_t, and\ \breve{x}_t$) and updates its parameters θ to produce a better corresponding mask for each sample. Accordingly, the learning process is attained by performing gradient-based approximate minimization on nested negative log-likelihood, as defined in Equation (42). Minimizing nested negative log-likelihood proved to enhance the generalisability of various segmentation models as discussed in section 4.3.7. Note that the objective of $TV_{\emptyset}$ unit is to minimize Equation (36), meanwhile, the segmentation model learning objective is illustrated in Equation (42).

The machinery of the entire training process is visually explained in Figure 32. Essentially, the segmentation model $f(x, \hat{x}_t, \breve{x}_t; \theta)$ learns how to map training manifold to the prediction manifold. Notice that during the training phase, instances outside the training manifold are introduced to enhance the segmentation performance on future unseen instances. The original images as well as the produced out-of-domain images have the same output mask which in turn act as implicit regularization mechanism to the mapping function $f(\cdot)$.

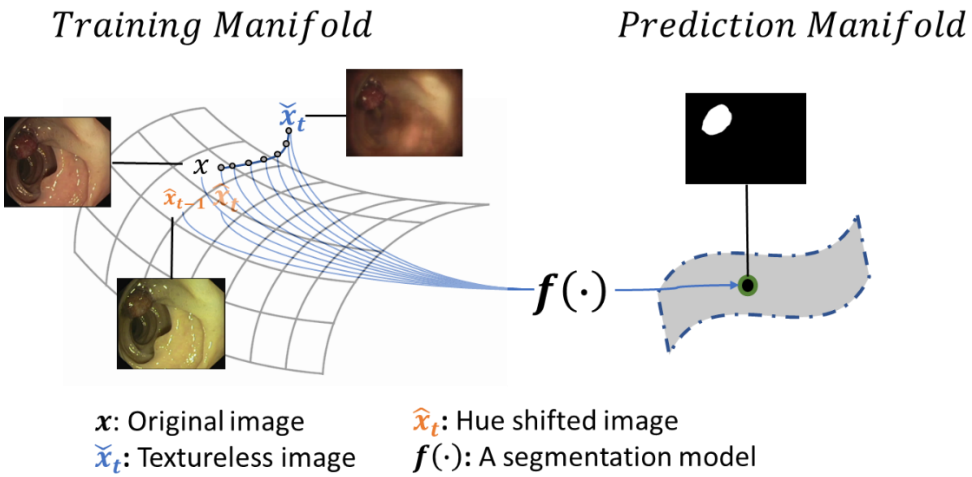


Figure 32 Hypothetical illustration of the proposed framework with respect to the training manifold.

The loss of the overall deep learning model can be defined by consulting Equation (36) and Equation (42) for $TV_{\emptyset}$ and the segmentation model objectives, respectively. For an epoch t, an original input image x, Hue shifted image $\hat{x}_t$, $TV_{\emptyset}$ image $\breve{x}_t$, and a corresponding image mask y, the loss function l is defined by:

$$\begin{aligned} l(x, \hat{x}_t, \breve{x}_t, y) = {} & l_{TV_{\emptyset}}(x, \breve{x}_t, y) + \\ & BCE(P_\theta(y|x), y) + \\ & BCE(P_\theta(y|\hat{x}_t), y) + \\ & BCE(P_\theta(y|\breve{x}_t), y) \end{aligned} \tag{43}$$

where $l_{TV_\emptyset}(\cdot)$ is the loss function of the $TV_\emptyset$ unit and $BCE(\cdot)$ is binary cross entropy loss for the segmentation model [139]. Finally, $P_\theta(\cdot \mid \cdot)$ is a conditional probability distribution which represents the output of the segmentation model. The $l_{TV_\emptyset}(\cdot)$ and $BCE(\cdot)$ are defined as follows:

$$l_{TV_\emptyset}(x, \check{x}_t, y) = \frac{\alpha}{HW} \|[x(w,h) - \check{x}_t(w,h)] \odot y\|^2 + \frac{\beta}{HW} \|\nabla \check{x}_t \odot (\mathbf{1} - y)\| \quad (44)$$

$$BCE(\tilde{y}, y) = \frac{1}{HW} sum[y \odot log\, \tilde{y} + (\mathbf{1} - y) \odot log(\mathbf{1} - \tilde{y})] \quad (45)$$

where $\odot$ is elementwise multiplication operator and $\mathbf{1} \in \mathbb{R}^{\boldsymbol{H \times W}}$ is a matrix of ones. The label y is a 2D mask image that has a pixel value of 1 for a polyp pixel and 0 otherwise. Finally, $log(\boldsymbol{B})$ and $sum[\boldsymbol{B}]$ are elementwise log function and a summation of elements of matrix $\boldsymbol{B} \in \mathbb{R}^{\boldsymbol{H \times W}}$, respectively.

4.2.3 Proposed model II: Spatial Interpolation

Following the previous presentation, another concrete instance of the proposed framework is presented in this subsection in which its core idea stemmed from Equation (42) (i.e., minimizing nested negative log-likelihood). Another image-to-image transformation is proposed by replacing the $TV_\emptyset$ with a texture interpolation unit $TI_\emptyset$. The overall proposed model is depicted in Figure 33.

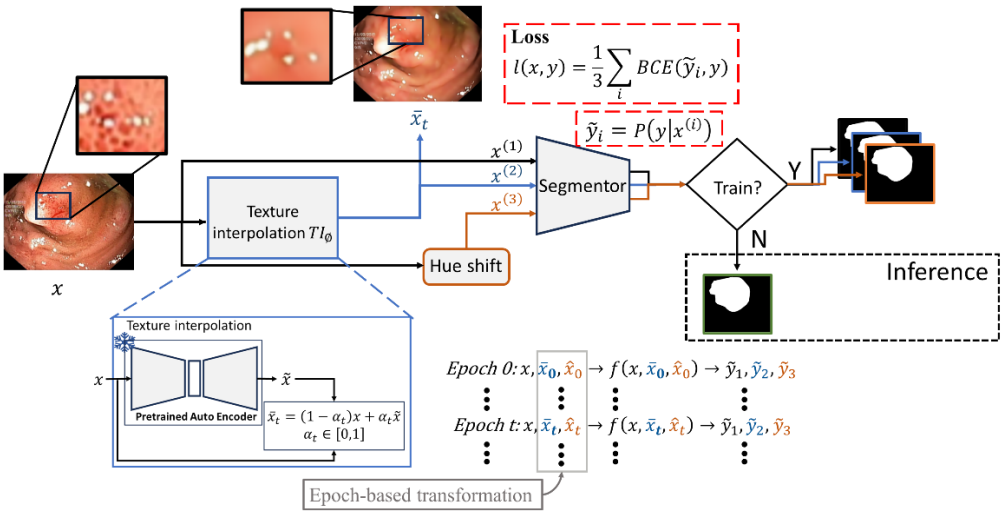


Figure 33 This figure depicts the second concrete implementation of the proposed framework. The input image is independently transformed by a random Hue-shift function and Texture Interpolation unit TI_ϕ.

In each training epoch, the proposed framework applies a texture interpolation transformation using $TI_\emptyset$ model to produce an image $\bar{x}$ that has different texture details than original inputs. The $TI_\emptyset$ get an input image and then applies a simple spatial interpolation between the input image x and a corresponding textureless version $\tilde{x}$, as depicted in Figure 33 and Figure 34.

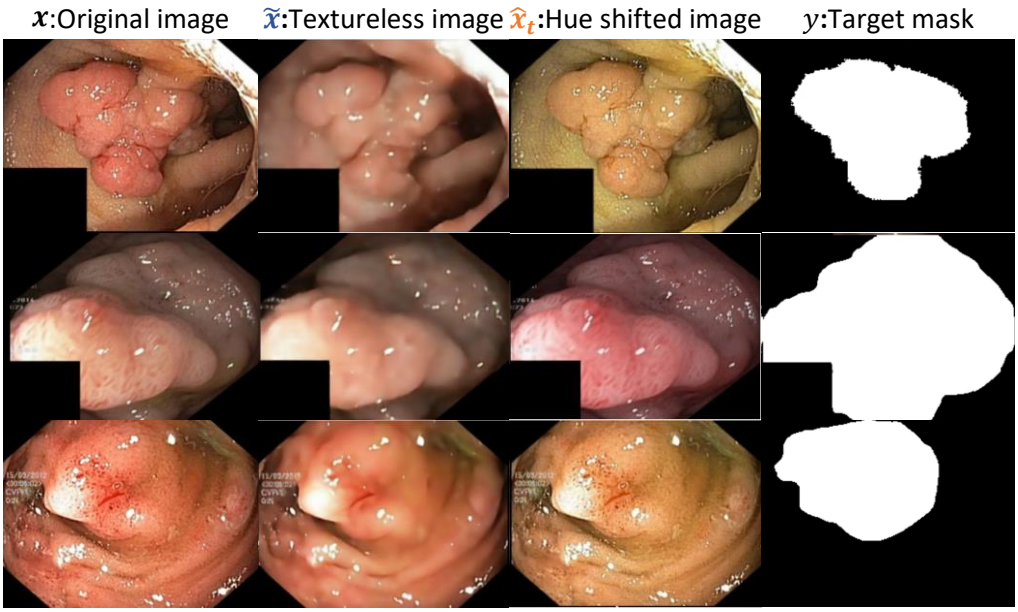


Figure 34 Some examples for input images and their corresponding transformations.

The texture interpolation is defined as follows:

$$\bar{x}_t = (1 - \alpha_t)x + \alpha_t\tilde{x}, \qquad \alpha_t \in [0,1] \tag{46}$$

$$\alpha_t = 0.5 \cdot \left(1 + cos\left(\frac{cycles \cdot 2\pi \cdot t}{T}\right)\right) \tag{47}$$

where $x, \bar{x}_t, and\ \tilde{x}$ are original image, interpolated image, and textureless image. Meanwhile α_t is a scalar that controls the interpolation rate. The two variables t and T are the epoch number and the total training epochs, respectively. Finally, $cycles$ is an integer that sets the interpolation periodicity rate between the original image x and its corresponding textureless image $\tilde{x}$, a visual illustration is depicted in Figure 35.

The textureless images $\tilde{x}$ are generated by a pre-trained autoencoder, which was trained beforehand, to approximately reconstruct the input image, preserving only the most relevant aspects of the images. Examples of textureless images produced by the auto-encoder are shown in Figure 34. The autoencoder's latent space has a relatively smaller spatial dimension compared to the encoding space, allowing it to capture essential features while omitting texture details, as shown in Figure 36. Additionally, the autoencoder was trained to generate polyp masks along with reconstructing original images to prevent the autoencoder from learning an identity function. Eventually objective loss for the autoencoder AE is defined as follows:

$$l_{AE}(x, \tilde{x}, y, \tilde{y}) = \|x - \tilde{x}\| + BCE(y, \tilde{y}) \tag{48}$$

where $x, \tilde{x}, y, and\ \tilde{y}$ are an input image, approximated textureless image, original mask, and output mask by the Autoencoder AE.

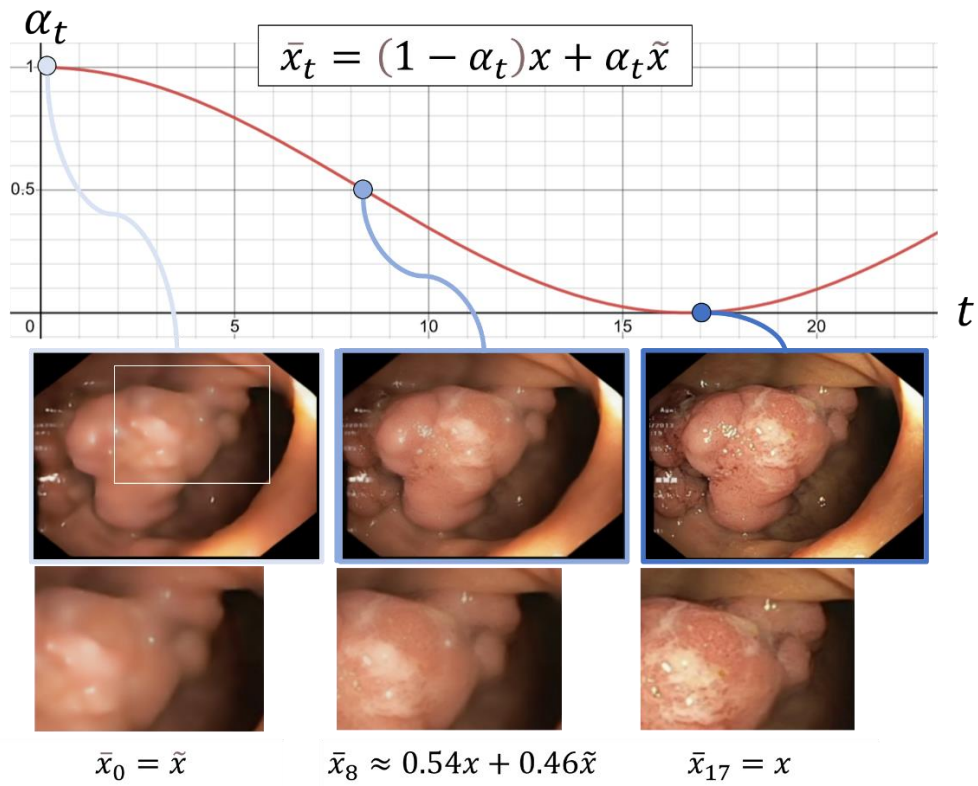


Figure 35 Interpolation between an original image x and a corresponding textureless version $\tilde{x}$. The x-axis of the curve represents the epoch number t, meanwhile, the y-axis represents interpolation rate α_t.

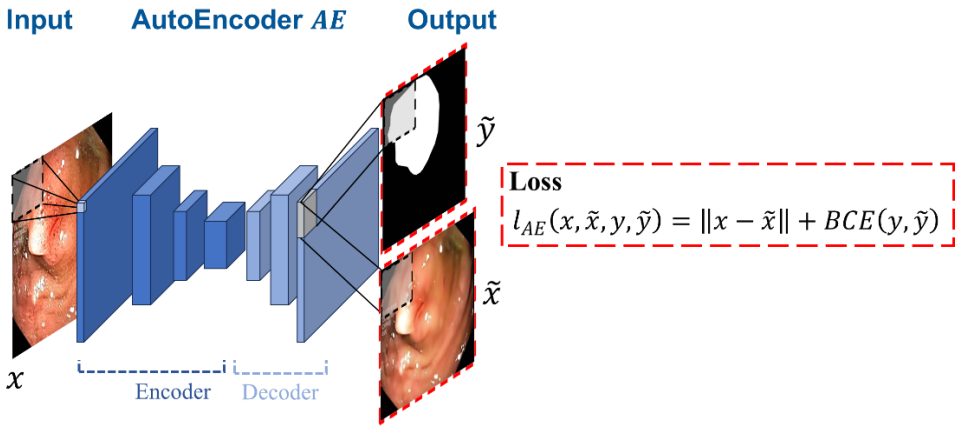


Figure 36 Autoencoder used to produce textureless approximation of the original image.

After the Autoencoder AE is trained, its weights are frozen and used to produce textureless version of the input image. The $TI_{\emptyset}$ unit then applies linear interpolation between the input image x and the textureless version $\tilde{x}$ to produce an image with different texture details, as depicted in Figure 35. The transformation depends on the epoch number t and, consequently, the scalar α_t.

For training a segmentation model, the same training strategy as the previous section is followed here as well. In each epoch, the segmentation model receives an input image with a different level of texture details to make the segmentation model invariant to texture changes, as shown in Figure 33 and Figure 35. Additionally, to enhance the model's robustness to colour changes, random hue transformations are applied. Accordingly, the loss for the segmentation model is defined as follows:

$$l(x, \hat{x}_t, \bar{x}_t, y) = BCE(P_\theta(y|x), y) + BCE(P_\theta(y|\hat{x}_t), y) + BCE(P_\theta(y|\bar{x}_t), y) \quad (49)$$

where $BCE(\cdot)$ is binary cross entropy loss for the segmentation model and it is defined in Equation (45). Meanwhile, $P_\theta(\cdot \mid \cdot)$ is a conditional probability distribution which represents the output of the segmentation model. Finally, $x, \hat{x}_t, \bar{x}_t, and\ y$ are original image, Hue shifted version, texture image-to-image transformed version, and target mask image, respectively.

The underlying concept behind this proposed design is to enable the segmentation model to learn from both the original training manifold and a corresponding, yet distinct, training manifold, as depicted in Figure 37. This approach helps mitigate overfitting issues that may arise due to deprived training samples. By including three different images within the same batch, learnable weights θ of the segmentation model are updated to capture robust features, resulting in improved results, as demonstrated in the experiment section. A segmentation model $f(x, \hat{x}_t, \bar{x}_t; \theta)$ is trained to map various samples with different texture and colour patterns to a single point on the prediction manifold (i.e., polyp mask), as depicted in Figure 37. During the training phase, $TI_\emptyset$ unit gradually shift original input from the training manifold to a new training manifold, meanwhile, make segmentation model $f(\cdot)$ to map these transformed samples to a single point on the prediction manifold, as seen in Figure 37. Eventually, deep features are learned by the segmentation model $f(\cdot)$, hence, performing better on unseen test sets.

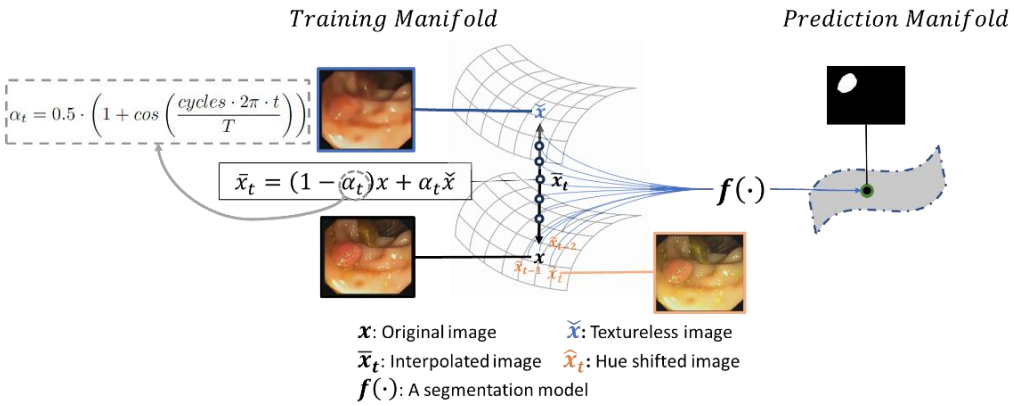


Figure 37 Hypothetical illustration of the proposed model with respect to the training manifold.

Mathematically, the training process can be interpreted as performing stochastic gradient descent on the following expectation:

$$-\mathbb{E}_{P_{data(x)}}\left[logP_\theta(y|x) + \mathbb{E}_{H(\hat{x}|x)}logP_\theta(y|\hat{x}) + \mathbb{E}_{TI_\emptyset(\bar{x}|x)}logP_\theta(y|\bar{x})\right] \quad (50)$$

where $P_{data(x)}$ is the training distribution, $H(\hat{x}|x)$ represents a conditional distribution over Hue shifted samples $\hat{x}$, given a data sample x, and finally, $TI_\emptyset(\check{x}|x)$ represents a conditional distribution over transformed samples $\check{x}$, given a data sample x. Accordingly, the learning process is attained by performing gradient-based approximate minimization on nested negative log-likelihood, as defined in Equation (50). Minimizing nested negative log-likelihood proved to enhance the generalisability of various segmentation models as established in the experiments section.

4.2.4 Polyp dataset

Three different datasets are used in the experiments, namely, Kvasir-SEG [102], CVC-ClinicDB [92], ETIS-Larib [103], and CVC-EndoSceneStill [104]. Those datasets are publicly available and have variety number of polyp images as illustrated in Table 10. The resolutions of the images in these datasets are not standardized. Therefore, prior to being fed into the deep learning models, all images were resized to a uniform size of 216-by-288 pixels.

Table 10 The used datasets in the experiments

Database	Number of images	Resolution in pixels
Kvasir-SEG	1000	384×288
CVC-ClinicDB	612	332×487 to 1920×1072
ETIS-Larib	196	1224×966
CVC-EndoSceneStill	912	384×288 to 574×500
Total	**2,720**	**N/A**

The training and validation images are constructed randomly from CVC-clinicDB dataset with a split of 70% and 30% for training set and validation set, respectively. Meanwhile the entire Kvasir-SEG dataset along with ETIS-Larib and CVC-EndoSceneStill are preserved as unseen test set. Nevertheless, it is worth mentioning that EndoSceneStill dataset is a combination of CVC-ClinicDB and other dataset called CVC-ColonDB [104].

4.3 Results and discussions

4.3.1 The used hyperparameters and metrics

All the experiments in this study were conducted with the same hyperparameters. PyTorch framework was used with Nvidia Tesla P100-PCI GPU. The hyperparameters are listed in Table 11:

Table 11 Hyperparameters of the conducted experiments

Hyperparameter	Value
Learning rate	0.01
Batch size	7 images
Numpy pseudorandom seed	0
Pytorch pseudorandom seed	0
Images size	Height=217, Width=288
Training epochs	300
Optimizer	Adam [140]
Loss	Binary Cross Entropy

Prevalent metrics in the literature were used to quantify the model performance. Given two sets A and B, the used metrics are listed as follows:

Intersection over Union (IoU)/Jaccard:

$$Jaccard = IoU = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP + FP + FN} \quad (51)$$

Dice/F1-score:

$$F1 = Dic = \frac{2|A \cap B|}{|A \cup B| + |A \cap B|} = \frac{2TP}{2TP + FP + FN} \quad (52)$$

Mean IOU (mIoU):

$$mIoU = \frac{IoU_{polyp} + IoU_{background}}{2} = \frac{TP}{2(TP + FP + FN)} + \frac{TN}{2(TN + FP + FN)} \quad (53)$$

Accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (54)$$

Precision:

$$Precision = \frac{TP}{TP + FP} \quad (55)$$

Recall:

$$Recall = \frac{TP}{TP + FN} \quad (56)$$

where TP, TN, FP, and FN are defined as follows:

TP (True Positive): A polyp pixel is correctly predicted as a polyp pixel.

TN (True Negative): A background pixel is correctly predicted as a background pixel.

FP (False Positive): A background pixel is wrongly predicted as polyp pixel.

FN (False Negative): A polyp pixel is wrongly predicted as a background pixel.

4.3.2 Subtle overestimation in the literature

Generally, when a deep learning model attains a high Intersection over Union (IoU) score for polyps, it suggests that the model effectively delineates polyp boundaries. Numerous studies have demonstrated this level of performance on publicly available datasets like CVC-ClinicDB [92]. However, it is important to question whether such accomplishments truly reflect the models' ability to perform well on unseen images in the future. These achievements are more likely a result of overestimation in the tested models rather than a sign of strong generalisation capabilities.

Considering that public datasets like CVC-ClinicDB [92] consist of sub-sequences of consecutive frames, it is inevitable that there will be repetitive images with potentially minor variations in lighting, angles, or focus. Consequently, if one were to randomly shuffle such a dataset and split it into different subsets, it would result in each subset containing similar-looking images, as depicted in Figure 38. Deep learning, known for its capability to overfit to training data, would then be evaluated on data that bears resemblance to what it has already seen during training. Consequently, deep learning models are likely to yield high performance, especially if they have overfit patterns present in the training set.

Under these circumstances, a deep learning model will acquire knowledge of the data patterns present in the training images, and as a result, it will accurately identify those same patterns in the validation images because both sets of images exhibit substantial similarities. Indeed, this exaggeration issue are empirically showcased in various deep learning models when evaluating them with the CVC-ClinicDB [92] dataset, as depicted in Figure 39.

Two distinct methods were applied to generate training and validation sets. The initial approach involved random shuffling of the dataset followed by splitting it into training and validation sets. The second approach was sequence-based, wherein a series of images were exclusively assigned to either the training or validation set. Consequently, two distinct dataset partitions were obtained. Subsequently, the best validation results for four distinct deep learning models were recorded, as depicted in Table 12.

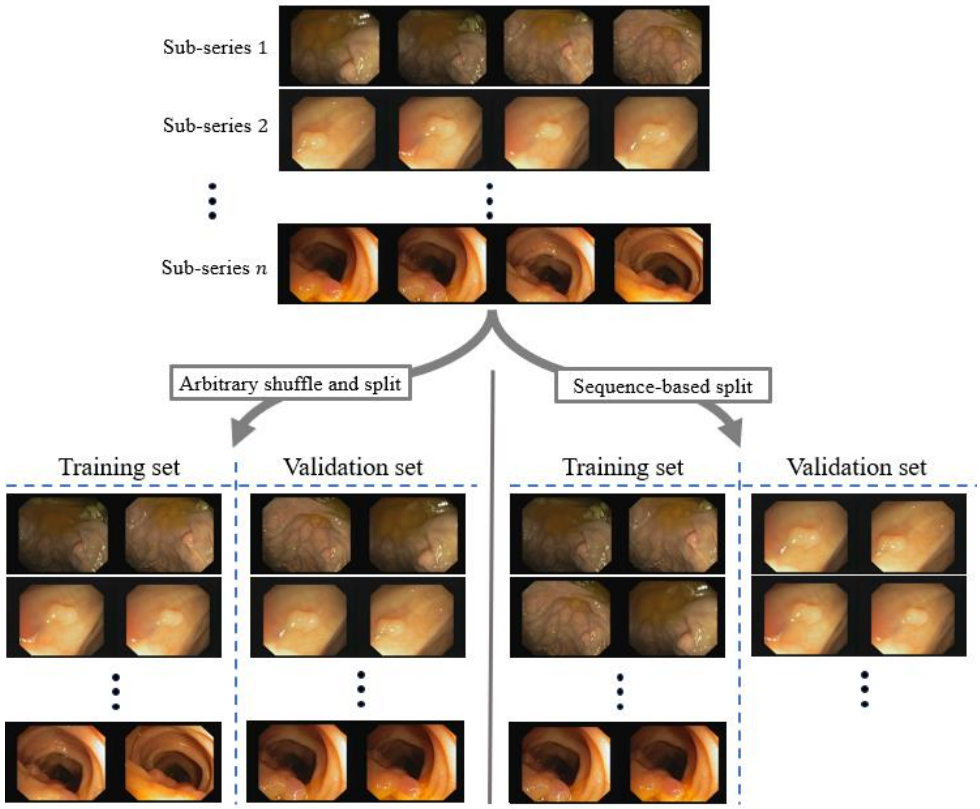


Figure 38 Partitioning a dataset comprising sequences of almost consecutive frames in an arbitrary manner will lead to two similar sets.

Table 12 Validation results using CVC-ClinicDB dataset given two approaches, Arbitrary and Series-based.

Model	**Split type**	
	Arbitrary	**Series-based**
	Polyp IoU	Polyp IoU
Unet	0.71268	0.41857
FCN	0.75312	0.44938
Lraspp	0.76834	0.45119
DeeplabV3	0.72453	0.38111

Regardless of the model employed, it is evident that random shuffling of CVC-ClinicDB leads to deep learning models achieving impressive validation Intersection over Union (IoU) scores. Conversely, when employing an informed splitting method, such as the sequence-based approach, a substantial decrease in validation performance becomes apparent, as illustrated in Table 12 and Figure 39. Both Table 12 and Figure 39 convey identical information, albeit the latter visually represents the results' patterns.

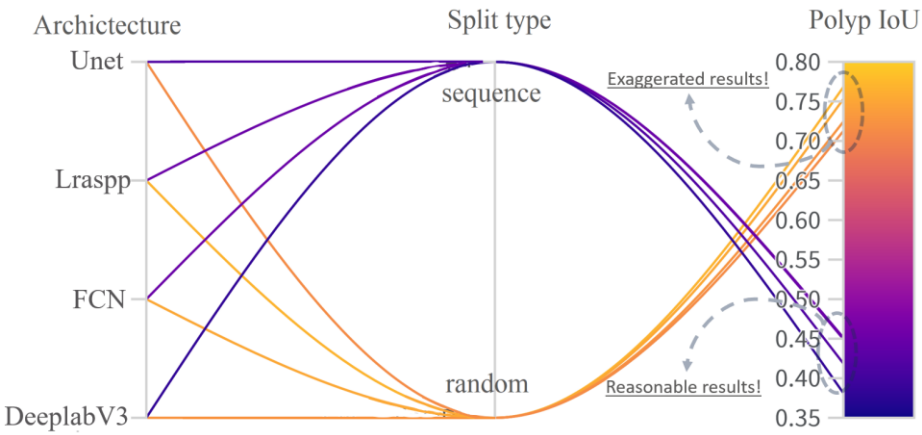


Figure 39 Randomly shuffling and splitting CVC-ClinicDB will result in an overestimation.

As evident, the choice of split method significantly influences the validation results obtained by any deep learning model. Consequently, it is common to observe substantial disparities in outcomes across various published works that utilise the same dataset. Attaining high validation results does not inherently imply a superior deep learning architecture. Hence, direct comparisons based solely on published results are not advisable. This subtle issue appears to persist across numerous research papers, highlighting the need for clarification in addressing it.

A more effective approach involves training and validating a model on one dataset and subsequently testing it on an entirely different dataset collected from a distinct healthcare source. This is the approach that was employed to conduct experiments and compare against the state-of-the-art models in this chapter. As detailed in the following section, CVC-ClinicDB was utilised for training and validation, and then assessed the performance of deep learning models on Kvasir-SEG [102], CVC-EndoSceneStill [104], and ETIS-Larib [103]. This methodology enables us to assess the generalisability of the proposed model and benchmark it against state-of-the-art models. Improved generalisability performance indicates a model's better suitability for clinical applications.

4.3.3 The effectiveness of the proposed framework using conventional segmentation models

In this subsection, the efficacy of employing the proposed framework in conjunction with traditional segmentation models will be illustrated. Both proposed models (i.e., $TV_{\emptyset}$ and $TI_{\emptyset}$) are compared against various segmentation models. The used conventional segmentation models include Unet [8], DeeplabV3 [89], FCN [75], and Lraspp [141]. It's worth noting that Unet and Lraspp are both considered lightweight models when compared to DeeplabV3 and FCN. For the purpose of training and validation, the CVC-ClinicDB dataset [92] was exclusively utilised, as seen in Table 13. The training and validation split is 70% and 30%, respectively. Other training to validation ratios could be used such as 80% to 20% or even 90% to 10%, however, our aim is to increase the chances of having a representative validation set by increasing validation samples. Conversely, the Kvasir-SEG [102], CVC-EndoSceneStill [104], and

ETIS-Larib [103] datasets were reserved for testing, as seen in Table 14, Table 15, and Table 16.

This approach of using the entire dataset solely for testing is the best strategy for assessing the generalisability of deep learning models, as these datasets originate from different medical centers. Thus, the testing datasets are treated as unseen sets, rather than combining all datasets and then partitioning them into training, validation, and test sets. However, there is an exception regards CVC-EndoSceneStill being not entirely from a different datacenter since part of it contains images from CVC-ClinicDB dataset and the other part from CVC-ColonDB [104] dataset.

For each of the four models, experiments were conducted to demonstrate the improvement in generalisability when applied to unseen test sets, both with and without the inclusion of the proposed framework. Additionally, each model's performance were evaluated with and without the utilisation of pretrained weights from the COCO segmentation dataset [142]. It's worth noting that the COCO dataset comprises non-medical images; however, pretrained weights were employed from this dataset to facilitate a comparative assessment of transfer learning against the proposed framework.

In general, there are moderate differences in metrics, in favor of the proposed framework, between the conventional deep learning models with and without the proposed framework on the validation set. In fact, the differences in polyp IoU range from approximately 0.3% to 5%. In certain instances, the proposed model attained superior validation polyp Intersection over Union (IoU); however, in other cases, conventional segmentation models outperformed only the proposed $TV_{\emptyset}$, as evident in Table 13. Such variations are unsurprising since both the training and validation data are drawn from the same dataset. In fact, the results achieved by these models are mainly influenced by overfitting to the training images. This overfitting had a negative impact on the generalisability of the conventional models when tested using unseen test sets, as established in Table 14, Table 15, and Table 16.

While the validation set results do not exhibit a clear pattern, a distinct advantage in favour of the proposed framework becomes apparent across all unseen test sets, as indicated in Table 14, Table 15, and Table 16. Apart from a few instances in the Recall results, the proposed framework consistently outperformed the corresponding segmentation models. This underscores the notion that irrespective of the chosen segmentation model, the use of the proposed framework consistently enhances the generalisability capabilities of the model.

Table 13 Validation results on the CVC-ClinicDB dataset. The training and validation split is 70% and 30%, respectively. The highest results are highlighted.

CVC-ClinicDB Validation set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Vanilla	Unet	93.88	43.56	56.13	68.33	57.21	68.47
	Proposed TV (Seg:Unet)	94.65	43.25	54.98	55.05	65.21	68.79
	Proposed TI (Seg:Unet)	**94.77**	**47.03**	**59.97**	**70.78**	**59.87**	**70.69**
	DeeplabV3	**94.86**	46.03	57.25	67.35	56.98	70.23
	Proposed TV (Seg:DeeplabV3)	94.45	39.12	49.55	53.61	52.15	66.6
	Proposed TI (Seg:DeeplabV3)	94.21	**46.95**	**58.66**	**68.2**	**58.76**	**70.38**
	FCN	**95.79**	45.29	55.84	60.9	57.85	70.37
	Proposed TV (Seg:Fcn)	94.97	45.94	56.3	64.8	59.2	70.26
	Proposed TI (Seg:Fcn)	95.35	**50.07**	**61.14**	**68.62**	**63.57**	**72.52**
	Lraspp	95.37	52.06	63.27	72.19	63.3	73.52
	Proposed TV (Seg:Lraspp)	96.23	53.97	64.46	70.39	**67.76**	74.95
	Proposed TI (Seg:Lraspp)	**96.32**	**57.83**	**68.33**	**77.42**	67.22	**76.91**
TL-Coco*	DeeplabV3	**94.31**	37.83	46.84	52.7	48.34	65.88
	Proposed TV (Seg:DeeplabV3)	95.28	**42.66**	52.57	52.27	**59.39**	**68.82**
	Proposed TI (Seg:DeeplabV3)	93.3	42.14	**53.68**	**68.22**	51.21	67.5
	FCN	**95.25**	47.81	58.96	66	61.71	71.33
	Proposed TV (Seg:Fcn)	94.49	40.24	50.93	57.52	55.08	67.17
	Proposed TI (Seg:Fcn)	95.07	**48.41**	**59.34**	**66.02**	**63.99**	**71.56**
	Lraspp	95.88	51.57	63.12	68.29	66.7	73.56
	Proposed TV (Seg:Lraspp)	96.22	52.87	63.54	66.08	67.85	74.4
	Proposed TI (Seg:Lraspp)	**96.33**	**55.12**	**66**	**69.68**	**70.05**	**75.59**
TL-Coco*: Segmentation model is pretrained on COCO dataset [142] and fine-tuned on polyp dataset CVC-ClinicDB.							

Table 14 Results of the test set Kvasir-Seg. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

Kvasir-SEG Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Vanilla	Unet	69.85	27.94	40.67	**72.56**	36.44	46.89
	Proposed TV (Seg:Unet)	**84.88**	**33.65**	**47.37**	51.06	**57.11**	**58.53**
	Proposed TI (Seg:Unet)	77.95	30.68	44.61	67.35	40.02	52.93
	DeeplabV3	76.66	23.95	34.84	50.53	34.74	49.24
	Proposed TV (Seg:DeeplabV3)	85.73	28.93	40.16	38.25	59.89	56.87
	Proposed TI (Seg:DeeplabV3)	**86.74**	**37.49**	**49.96**	**50.69**	**62.46**	**61.51**
	FCN	72.42	29.79	41.85	**68.46**	38.42	49.45
	Proposed TV (Seg:Fcn)	**87.67**	**42.29**	**53.62**	55.05	**66.55**	**64.4**
	Proposed TI (Seg:Fcn)	87.29	40	52.57	53.01	64.46	63.04
	Lraspp	85.35	36.43	48.64	53.95	55.44	60.19
	Proposed TV (Seg:Lraspp)	89.66	50.69	62.04	57.9	**79.22**	69.69
	Proposed TI (Seg:Lraspp)	**89.84**	**50.99**	**62.22**	**59.61**	77.86	**69.96**
TL-Coco*	DeeplabV3	83.16	19.3	28.13	28.72	42.58	50.8
	Proposed TV (Seg:DeeplabV3)	**87.07**	**35.63**	46.93	43.08	**69.63**	**60.93**
	Proposed TI (Seg:DeeplabV3)	86.11	34.97	**47.17**	**47.83**	59.15	59.97
	FCN	76.46	30.93	43.11	**62.73**	44.62	52.36
	Proposed TV (Seg:Fcn)	87.05	35.33	47.1	43.49	**68.26**	60.76
	Proposed TI (Seg:Fcn)	**87.29**	**39.57**	**51.61**	51.92	64.65	**62.84**
	Lraspp	84.52	34.33	46.63	50.05	56.85	58.72
	Proposed TV (Seg:Lraspp)	88.26	41.34	51.43	46.68	74.78	64.45
	Proposed TI (Seg:Lraspp)	**89.48**	**48.32**	**59.12**	**55.73**	**77.88**	**68.47**
TL-Coco*: Segmentation model is pretrained on COCO dataset [142] and fine-tuned on polyp dataset CVC-ClinicDB.							

Table 15 This is the results of the test set CVC_EndoSceneStill. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

CVC_EndoSceneStil Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Vanilla	Unet	92.13	46.99	56.79	65.73	62.17	69.36
	Proposed TV (Seg:Unet)	**95.22**	52.1	62.37	62.17	**75.29**	73.52
	Proposed TI (Seg:Unet)	94.65	**54.25**	**64.33**	**71.01**	66.46	**74.27**
	DeeplabV3	94	56.87	65.07	**74.41**	65.9	75.27
	Proposed TV (Seg:DeeplabV3)	**95.95**	56.49	65.33	67.28	**69.7**	76.1
	Proposed TI (Seg:DeeplabV3)	95.12	**60.44**	**69.09**	**75.97**	69.65	**77.64**
	FCN	93.27	53.53	61.9	70.54	63.64	73.21
	Proposed TV (Seg:Fcn)	96.06	60.54	69.41	74.31	71.24	78.16
	Proposed TI (Seg:Fcn)	**96.62**	**64.64**	**72.34**	**74.94**	**75.02**	**80.52**
	Lraspp	96.29	61.89	70.2	75.22	70.96	78.97
	Proposed TV (Seg:Lraspp)	97.47	70.62	78.22	80.8	**80.43**	83.95
	Proposed TI (Seg:Lraspp)	**97.6**	**71.16**	**78.68**	**82.82**	78.95	**84.29**
TL-Coco*	DeeplabV3	94.03	45.77	54.29	59.43	57.3	69.75
	Proposed TV (Seg:DeeplabV3)	**96.58**	**60.95**	**68.8**	68.9	**74.01**	**78.65**
	Proposed TI (Seg:DeeplabV3)	95.1	57.18	65.97	**75.98**	63.16	76
	FCN	93.94	54.75	63.11	69.78	65.79	74.18
	Proposed TV (Seg:Fcn)	95.89	57.69	66.58	69.31	71	76.66
	Proposed TI (Seg:Fcn)	**96.28**	**63.03**	**70.7**	**74.6**	**73.23**	**79.55**
	Lraspp	96.2	58.18	68.19	69.41	73.59	77.05
	Proposed TV (Seg:Lraspp)	97.39	66.15	74.37	74.53	**79.38**	81.67
	Proposed TI (Seg:Lraspp)	**97.4**	**68.04**	**75.29**	**77.76**	77.89	**82.63**
TL-Coco*: Segmentation model is pretrained on COCO dataset [142] and fine-tuned on polyp dataset CVC-ClinicDB.							

Table 16 This is the results of the test set ETIS_LaribPolypDB. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

ETIS-LaribPolypDB Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Vanilla	Unet	51.35	7.77	12.91	**73.16**	9.09	28.78
	Proposed TV (Seg:Unet)	**94.1**	**11.46**	**16.74**	17.92	**23.05**	**52.71**
	Proposed TI (Seg:Unet)	69.82	9.41	15.44	57.01	11.82	39.2
	DeeplabV3	89.2	7.86	12.08	24.25	11.77	48.42
	Proposed TV (Seg:DeeplabV3)	**93.37**	13.95	19.47	**28.08**	**22.74**	**53.58**
	Proposed TI (Seg:DeeplabV3)	91.77	**14.31**	**19.71**	26.28	22.29	52.94
	FCN	60.05	9.01	14.16	**62.68**	11.3	34.05
	Proposed TV (Seg:Fcn)	**94.04**	**16.63**	**21.81**	28.22	**26.99**	**55.28**
	Proposed TI (Seg:Fcn)	93.47	11.47	16.42	20.75	20.34	52.39
	Lraspp	91.1	11.03	16.17	22.95	18.04	50.95
	Proposed TV (Seg:Lraspp)	95.53	20.45	25.6	26.11	31.45	57.93
	Proposed TI (Seg:Lraspp)	**95.55**	**23.44**	**30.54**	**32.89**	**35.96**	**59.42**
TL-Coco*	DeeplabV3	93.36	6.9	10.63	15.03	11.45	50.04
	Proposed TV (Seg:DeeplabV3)	94.13	**16.62**	**21.67**	**25.18**	**28.29**	**55.32**
	Proposed TI (Seg:DeeplabV3)	**95.49**	11.63	16.18	15.79	24.63	53.49
	FCN	57.29	7.64	12.52	**64.43**	9.49	31.88
	Proposed TV (Seg:Fcn)	**94.22**	**14.58**	**20.42**	23.58	**24.43**	**54.34**
	Proposed TI (Seg:Fcn)	93.06	13.04	18.18	24.32	20.7	52.98
	Lraspp	85.41	13.4	19.2	**34.92**	19.7	49.22
	Proposed TV (Seg:Lraspp)	**95.68**	**17.63**	22.66	22.26	31.71	**56.6**
	Proposed TI (Seg:Lraspp)	95.51	16.89	**22.72**	21.82	**34.5**	56.15
TL-Coco*: Segmentation model is pretrained on COCO dataset [142] and fine-tuned on polyp dataset CVC-ClinicDB.							

In certain instances, the recall results of deep learning models surpass those of the proposed framework. It's important to note that achieving high recall results does not necessarily signify excellent performance but rather evokes a bias towards polyp pixels. When there's a high recall coupled with relatively lower Precision, it indicates that a model exhibits a higher sensitivity in classifying background pixels as polyp pixels. Consequently, it amplifies the false-positive errors, resulting in lower precision. In fact, even a simple, naive model that labels all pixels as polyp would achieve a perfect 100% recall rate at the expense of the precision.

Notably, what's interesting is that the proposed framework, when paired with any segmentation model, outperforms its corresponding model with transfer learning, as illustrated in Figure 40. In fact, this is true not only for the proposed $TV_{\emptyset}$ model, but rather both the proposed $TV_{\emptyset}$ and $TI_{\emptyset}$ models, as shown in Figure 40. In this specific scenario, transfer learning did not appear to provide any added benefit. The rationale behind this lies in the fact that these segmentation models were originally trained on natural images, such as the COCO dataset, which significantly differs from the context of colonoscopy images. It is reasonable to anticipate a substantial performance improvement if these models were pretrained on colonoscopy images instead of non-medical datasets. In fact, there is currently a growing call for the development of an ImageNet-like dataset tailored specifically for endoscopy.

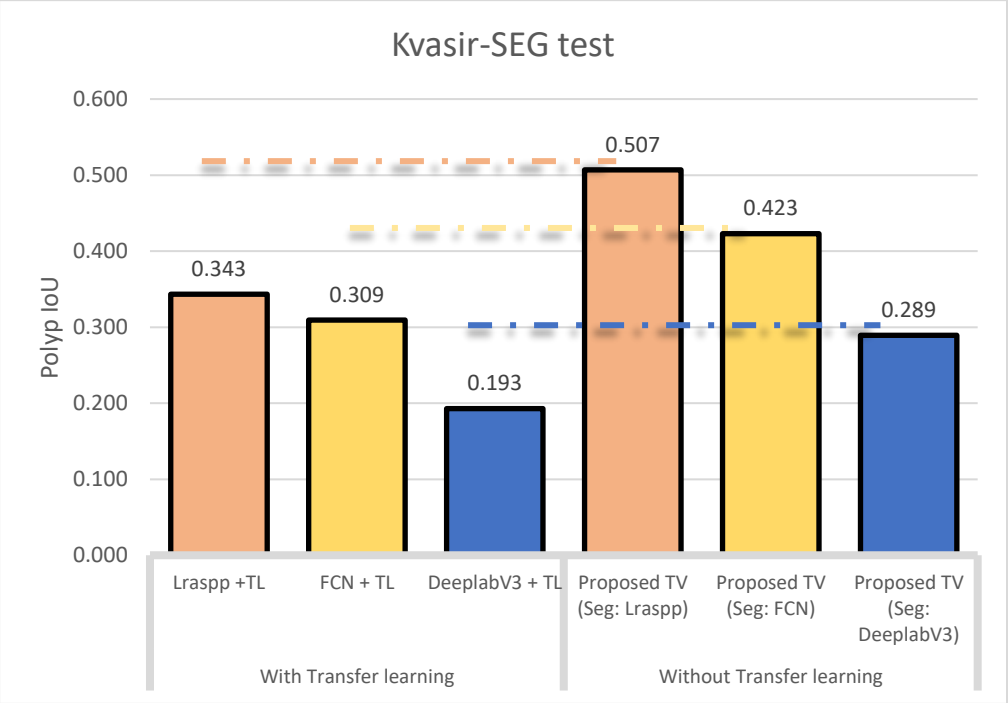


Figure 40 Polyp Intersection over Union results on Kvasir-SEG (i.e., test set). The proposed segmentation models without TL against corresponding segmentation model with TL.

4.3.4 Proposed framework against the state-of-the-art models

The preceding section demonstrated how the integration of the proposed framework led to a notable improvement in the generalisability performance of traditional segmentation models. In this section, a comparative analysis of the proposed framework will be conducted against recent models explicitly designed for polyp segmentation. These models include ACSNet [98] (2020), MSNet [125] (2021), CaraNet [99] (2022), and M^2SNet [127] (2023).

ACSNet was introduced with the aim of capturing both global context features and local context information through an encoder-decoder framework that employs adaptive context selection [98]. In contrast, MSNet and M2SNet are U-Net-based architectures that incorporate multiscale subtraction units between the encoder and decoder, rather than relying solely on direct skip connections [127]. For a more detailed understanding of segmentation architectures, refer to section 2.4. On the other hand, CaraNet leverages a self-attention mechanism and a channel-wise feature pyramid module to extract feature information from small medical objects [99].

Similar recall results are evident on the validation set, though, the proposed framework overcome the state-of-the-art models on other metrics, as illustrated in Table 17. For example, the discrepancy in recall between the proposed framework $TV_{\emptyset}$ with Lraspp and MSNet is just 0.51%. Conversely, MSNet negligibly outperforms the proposed framework $TV_{\emptyset}$ with FCN in terms of dice scores. However, a substantial performance gap emerges strongly in favour of the proposed framework when tested on unseen datasets, as depicted in Table 18, Table 19, and Table 20.

Overall, the proposed framework consistently outperformed state-of-the-art models (SOTA) across all metrics on every test set. Specifically, reflecting on the Kvasir-SEG test set and considering the polyp IoU metric for both segmentation networks (FCN and Lraspp), the proposed framework surpassed other SOTA models by a notable margin, ranging from approximately 10.62% to 29.82%. This clearly underscores the effectiveness of the proposed framework on unseen test datasets, highlighting its superior generalisability capabilities.

Table 17 Validation results on the CVC-ClinicDB dataset. The training and validation split is 70% and 30%, respectively. The highest two results are highlighted.

CVC-ClinicDB Validation set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
TL*	MSNet	94.78	45.43	56.45	69.88	56.27	69.91
	M^2SNet	90.33	33.08	45.2	66.04	42.57	61.35
	CaraNet	91.52	28.53	38.52	53.09	35.5	59.76
	ACSNet	88.0	29.73	41.74	67.9	37.32	58.54
w/o TL	Proposed TV (Seg:FCN)	94.97	45.94	56.3	64.8	59.2	70.26
	Proposed TI (Seg:FCN)	95.35	50.07	61.14	68.62	63.57	72.52
	Proposed TV (Seg:Lraspp)	**96.23**	**53.97**	**64.46**	**70.39**	**67.76**	**74.95**
	Proposed TI (Seg:Lraspp)	**96.32**	**57.83**	**68.33**	**77.42**	**67.22**	**76.91**
TL*: Only the encoder is pretrained using ImageNet.							

Table 18 Test results of the Kvasir-Seg dataset. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

Kvasir-SEG Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
TL*	MSNet	85.23	29.21	39.95	41.11	55.37	56.65
	M2SNet	68.69	25.85	37.87	**71.28**	31.3	45.25
	CaraNet	73.07	27.38	39.65	**66.17**	34.92	48.66
	ACSNet	84.28	20.87	30.29	31.99	43.64	52.1
w/o TL	Proposed TV (Seg:FCN)	87.67	42.29	53.62	55.05	66.55	64.4
	Proposed TI (Seg:Fcn)	87.29	40	52.57	53.01	64.46	63.04
	Proposed TV (Seg:Lraspp)	**89.66**	**50.69**	**62.04**	57.9	**79.22**	**69.69**
	Proposed TI (Seg:Lraspp)	**89.84**	**50.99**	**62.22**	59.61	**77.86**	**69.96**
TL*: Only the encoder is pretrained using ImageNet.							

Table 19 Test results of the CVC_EndoSceneStil dataset. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

CVC_EndoSceneStil Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
TL*	MSNet	94.44	52.65	61.54	69.64	62.66	73.39
	M2SNet	88.36	37.7	48.12	66.35	46.85	62.69
	CaraNet	90.06	29.98	39.3	50.39	40.33	59.78
	ACSNet	88.42	29.11	40.78	61.84	38.18	58.48
w/o TL	Proposed TV (Seg:FCN)	96.06	60.54	69.41	74.31	71.24	78.16
	Proposed TI (Seg:Fcn)	96.62	64.64	72.34	74.94	75.02	80.52
	Proposed TV (Seg:Lraspp)	**97.47**	**70.62**	**78.22**	**80.8**	**80.43**	**83.95**
	Proposed TI (Seg:Lraspp)	**97.6**	**71.16**	**78.68**	**82.82**	**78.95**	**84.29**
TL*: Only the encoder is pretrained using ImageNet.							

Table 20 Test results of the ETIS-LaribPolypDB dataset. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

ETIS-LaribPolypDB Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
TL*	MSNet	87.09	9.65	14.63	28.42	18.58	48.23
	M2SNet	58.06	8.55	13.52	**64.06**	9.55	32.64
	CaraNet	69.01	10.38	16.41	**66.06**	13.77	39.24
	ACSNet	93.22	7.84	11.47	17.37	12.7	50.44
w/o TL	Proposed TV (Seg:FCN)	94.04	16.63	21.81	28.22	26.99	55.28
	Proposed TI (Seg:Fcn)	93.47	11.47	16.42	20.75	20.34	52.39
	Proposed TV (Seg:Lraspp)	**95.53**	**20.45**	**25.6**	26.11	**31.45**	**57.93**
	Proposed TI (Seg:Lraspp)	**95.55**	**23.44**	**30.54**	32.89	**35.96**	**59.42**
TL*: Only the encoder is pretrained using ImageNet.							

Nevertheless, there are a few exceptions, particularly in the case of the recall metric in certain instances. For example, in Table 18, CaraNet has a recall that is approximately 6.56% higher than the proposed framework $TI_{\emptyset}$ with Lraspp. However, this elevated recall achieved by the CaraNet model comes at the expense of a lower precision result. It signifies a pronounced inclination towards the polyp class, resulting in the misclassification of a substantial background region as polyp. In essence, CaraNet is prone to high false-positive errors, which have a detrimental impact on its overall performance across other metrics. In fact, when comparing the proposed framework $TI_{\emptyset}$ with Lraspp and CaraNet in terms of precision and mean IoU (mIoU) on the unseen Kvasir-SEG test set, the difference stands at 42.94% and 21.3% in favour of the proposed framework, respectively. Similar comparisons are found with the other proposed frameworks $TV_{\emptyset}$.

Overall, state-of-the-art (SOTA) models did not attain high results on the unseen test sets primarily because of severe overfitting during training. In fact, as it will be showcased in the subsequent section, the generalisability performance of SOTA models can be significantly improved by introducing random augmentation to the training batches.

4.3.5 Proposed framework against the state-of-the-art models with augmentation

Limited training data increase the chances of training overfitting specially when training complex deep learning models, especially with limited training data. Several methods exist to lessen this problem, including utilising a regularization term to the loss function, incorporating dropout layers, applying early stopping, and inflating the dataset through augmentation. Given our constrained training dataset, augmentation holds the potential to be beneficial.

Consequently, the training batches were inflated by implementing random augmentation during the training process. Throughout the training epochs, random augmentation is applied to each training batch, and both the original and augmented images are provided as inputs to the segmentation model. Since the augmentation process is stochastic, varying degrees of augmentation are applied to the training images for each batch and during each training epoch. The augmentation techniques employed include random horizontal and vertical flipping, rotation, shearing, brightness adjustment, and hue shifting.

The state-of-the-art SOTA models significantly benefited from applying random augmentation during training and accordingly achieved better validation and test results as shown in Table 21, Table 22,

Table 23, and Table 24. Furthermore, Figure 41 shows performance of SOTA models on the unseen Kvasir-SEG test set with and without augmentation. Significant enhancements are observed after applying augmentation which indicates that SOTA models initially suffered from a severe training overfitting.

Table 21 Validation results on the CVC-ClinicDB dataset. The training and validation split is 70% and 30%, respectively. The highest two results are highlighted.

CVC-ClinicDB Validation set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Aug+TL*	MSNet	93.78	41.45	51.77	58.65	56.65	67.42
	M^2SNet	91.61	32.87	42.34	56.62	42.53	62
	CaraNet	93.59	36.7	48.45	58.91	50.41	64.95
	ACSNet	94.69	42.93	54.37	60.61	56.5	68.63
w/o Aug+TL*	Proposed TV (Seg:FCN)	94.97	45.94	56.3	64.8	59.2	70.26
	Proposed TI (Seg:FCN)	95.35	50.07	61.14	68.62	63.57	72.52
	Proposed TV (Seg:Lraspp)	**96.23**	**53.97**	**64.46**	**70.39**	**67.76**	**74.95**
	Proposed TI (Seg:Lraspp)	**96.32**	**57.83**	**68.33**	**77.42**	**67.22**	**76.91**

Aug+TL*: Only the encoder is pretrained using ImageNet. Random augmentation is applied to training batches on-the-fly. Augmentation used were random rotation, horizontal & vertical flipping, shear, brightness, and hue shifting.

Table 22 Test results of the Kvasir-Seg dataset. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

Kvasir-SEG Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Aug+TL*	MSNet	87.25	42.6	54.96	**59.98**	61.09	64.27
	M^2SNet	86.39	40.39	52.43	56.85	59.82	62.73
	CaraNet	84.57	32.03	44.53	50.35	50.66	57.66
	ACSNet	88.2	42.16	54.08	48.26	75.7	64.78
w/o Aug+TL*	Proposed TV (Seg:FCN)	87.67	42.29	53.62	55.05	66.55	64.4
	Proposed TI (Seg:Fcn)	87.29	40	52.57	53.01	64.46	63.04
	Proposed TV (Seg:Lraspp)	**89.66**	**50.69**	**62.04**	57.9	**79.22**	**69.69**
	Proposed TI (Seg:Lraspp)	**89.84**	**50.99**	**62.22**	**59.61**	**77.86**	**69.96**

Aug+TL*: Only the encoder is pretrained using ImageNet. Random augmentation is applied to training batches on-the-fly. Augmentation used were random rotation, horizontal & vertical flipping, shear, brightness, and hue shifting.

Table 23 Test results of the CVC_EndoSceneStil dataset. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

CVC_EndoSceneStill Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Aug+TL*	MSNet	96.18	61.28	69.31	73.96	69.98	78.61
	M^2SNet	93.17	50.52	59.04	69.86	57.15	71.67
	CaraNet	93.33	39.91	51.56	60.73	54.11	66.45
	ACSNet	96.51	60.67	69.43	70.86	73.17	78.47
w/o Aug+TL*	Proposed TV (Seg:FCN)	96.06	60.54	69.41	74.31	71.24	78.16
	Proposed TI (Seg:Fcn)	96.62	64.64	72.34	74.94	75.02	80.52
	Proposed TV (Seg:Lraspp)	**97.47**	**70.62**	**78.22**	**80.8**	**80.43**	**83.95**
	Proposed TI (Seg:Lraspp)	**97.6**	**71.16**	**78.68**	**82.82**	**78.95**	**84.29**

Aug+TL*: Only the encoder is pretrained using ImageNet. Random augmentation is applied to training batches on-the-fly. Augmentation used were random rotation, horizontal & vertical flipping, shear, brightness, and hue shifting.

Table 24 Test results of the ETIS-LaribPolypDB dataset. Models were trained and validated using CVC-ClinicDB dataset. The highest results are highlighted.

ETIS-LaribPolypDB Test set							
Settings	**Model**	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Aug+TL*	MSNet	94.24	10.56	13.8	14.7	17.1	52.34
	M^2SNet	94.51	21.13	26.9	32.53	26.82	57.73
	CaraNet	91.09	12.61	18.53	27.45	19.51	51.75
	ACSNet	95.43	**25.65**	**32.55**	**35.74**	**35.09**	**60.47**
w/o Aug+TL*	Proposed TV (Seg:FCN)	94.04	16.63	21.81	28.22	26.99	55.28
	Proposed TI (Seg:Fcn)	93.47	11.47	16.42	20.75	20.34	52.39
	Proposed TV (Seg:Lraspp)	**95.53**	20.45	25.6	26.11	31.45	57.93
	Proposed TI (Seg:Lraspp)	**95.55**	**23.44**	**30.54**	**32.89**	**35.96**	**59.42**

Aug+TL*: Only the encoder is pretrained using ImageNet. Random augmentation is applied to training batches on-the-fly. Augmentation used were random rotation, horizontal & vertical flipping, shear, brightness, and hue shifting.

Nevertheless, the proposed framework (i.e., $TV_{\emptyset}$ and $TI_{\emptyset}$ versions) with Lraspp segmentation model achieved better results than SOTA models with augmentation. However, unsignificant margin between SOTA with augmentation and the proposed framework with Fully Convolutional Network (FCN) model are present. FCN is known to have low resolution prediction which affect the accuracy of segmentation boundaries [143]. This coarse prediction by FCN is due to consecutive layers of convolution and pooling which results in having down-sampled output feature map. Nevertheless, both SOTA models and the proposed framework achieved high results on CVC_EndoSceneStill compared to other test sets. This is mainly due to the fact that CVC_EndoSceneStill combines two different datasets include CVC-ClinicDB which used as training and validation, and CVC_ColonDB which is considered wholeheartedly unseen. On the other hand, all evaluated segmentation models achieved relatively low results on ETIS-LaribPolypDB dataset. This is due to the discrepancy between the training/validation set (i.e., CVC-ClinicDB) and ETIS-LaribPolypDB in terms of polyp shapes and images' resolution. Furthermore, the author of ETIS-LaribPolypDB modified the raw images to delete its black borders produced by the colonoscopy devices which usually have meta information. This discrepancy between the training set and ETIS-LaribPolypDB set caused all tested deep learning models to achieve less than other test sets.

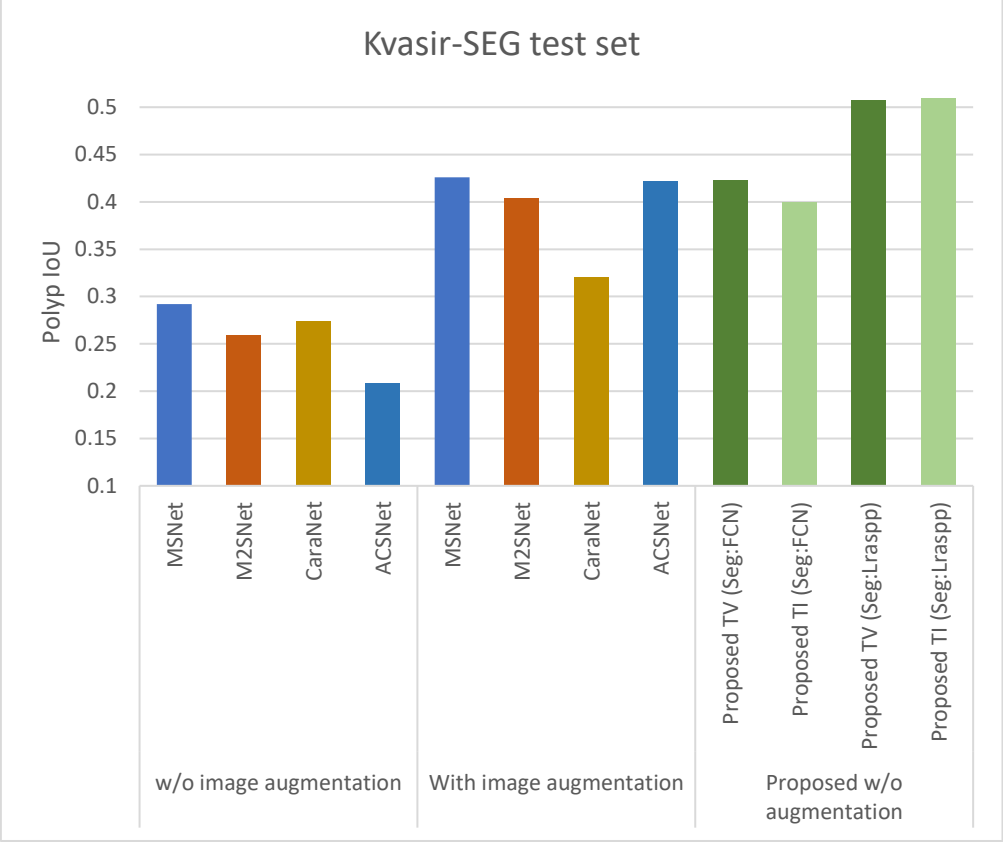


Figure 41 The effects of augmentation on the state-of-the-art models in comparison to the proposed framework without augmentation.

The proposed framework with Lraspp segmentation model achieved the state-of-the-art on all used test sets even though SOTA models utilised augmentation. A significant margin between the proposed model and SOTA models are observed which signals the generalisability effectiveness of the proposed framework.

4.3.6 Comparisons using EndoSceneStill benchmark

For further analysing the proposed framework, a standard benchmark based on EndoSceneStill dataset is selected [104]. Data split created by [104] was followed to be able to do fair comparisons against the proposed framework. The datasets are divided into training, validation and test set with 547, 183, and 182 images, respectively. The results on the test set are reported in Table 25 [104]. FCN8 model architecture was used in with two extra classes were added in the original EndoSceneStill benchmark including light specularity and lumen (i.e., the opening inside the bowels).

It is clear from Table 25 that the proposed deep learning framework achieved the state-of-the-art results on every metric with a noticeable margin. Since the background is the dominant class, all models achieved high background Intersection over Union (IoU) as opposed to polyp IoU. In terms of polyp IoU, the difference between the proposed framework and FCN8 with augmentation, which considered best results, is ~8% to ~12.9% in favour of the proposed framework. Given that the stated results are based on unseen test set, it is then evident that the proposed framework enhances the generalisability of segmentation models.

Table 25 This benchmark table is taken directly from [104]. This table presents results on the test set. Training, validation, and test set are all created from EndoSceneStill dataset and provided by [104].

	Augmentation	IoU background	IoU polyp	IoU lumen	IoU Specularity	IoU mean	Accuracy mean
4 classes	None	86.36	38.51	43.97	32.98	50.46	87.40
3 classes	None	84.66	47.55	36.93	N/A	56.38	86.08
2 classes	None	94.62	50.85	N/A	N/A	72.74	94.91
4 classes	Combination	88.81	51.60	41.21	38.87	55.13	89.69
State-of-the-art methods							
[144], [145], and [146]	N/A	73.93	22.13	23.82	44.86	41.19	75.58
Proposed TI (Seg:Lraspp)	None	**94.57**	**58.65**	N/A	N/A	**76.61**	**94.8**
Proposed TV (Seg:Lraspp)	None	**94.76**	**64.56**	N/A	N/A	**79.66**	**95.01**

4.3.7 Analysis of the proposed framework

Qualitative and quantitative analysis of the proposed framework are presented in this section by first present ablation study followed by demonstrating gradients of the transformed images by the $TV_{\emptyset}(\check{x}|x)$ and $TI_{\emptyset}(\bar{x}|x)$ components. The purpose of the ablation study is to explore the performance of the proposed framework by removing certain components to understand the enhancement of each component to the overall framework. Meanwhile, images' gradient help understand the effect of both $TV_{\emptyset}(\check{x}|x)$ on the input images in terms of their edges and textures.

Ablation study

There are mainly three components in the proposed framework including, a segmentation model (i.e., Lraspp), Hue transformation $H(\hat{x}|x)$, and texture transformation unit (i.e., Total Variational $TV_{\emptyset}(\check{x}|x)$ or Texture Interpolation $TI_{\emptyset}(\bar{x}|x)$). To understand the contribution of each component of the proposed framework $H(\hat{x}|x)$, $TV_{\emptyset}(\check{x}|x)$, or $TI_{\emptyset}(\bar{x}|x)$ transformations were alternatively removed and the generalisability performance on unseen Kvasir-SEG dataset was evaluated, as seen in Table 26. Consistent patterns, as per previous experiments, are prevalent in which segmentation models achieved comparable results on the validation set. Meanwhile they evolved disparity results on unseen Kvasir-SEG test set.

Table 26 Validation results on the CVC-ClinicDB dataset. The training and validation split is 70% and 30%, respectively. The highest records are highlighted.

CVC-ClinicDB Validation set						
Model	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Lraspp	95.37	52.06	63.27	72.19	63.3	73.52
Lraspp + Hue $H(\hat{x}|x)$	95.74	51.65	63.21	63.66	**69.85**	73.56
Lraspp + total variational $TV_{\emptyset}(\check{x}|x)$	95.53	52.33	63.01	74.03	61.66	73.73
Lraspp + $H(\hat{x}|x)$ + $TV_{\emptyset}(\check{x}|x)$	96.23	53.97	64.46	70.39	67.76	74.95
Lraspp + $H(\hat{x}|x)$ + $TI_{\emptyset}(\bar{x}|x)$	**96.32**	**57.83**	**68.33**	**77.42**	67.22	**76.91**

Vanilla Lraspp segmentation model along with its other variations achieved polyp IoU of 51.6% to 57.8%. However, the results' range is enlarged on the test set in which the lowest polyp IoU of 36.4% is achieved by Lraspp, and the highest polyp IoU of 50.9% is achieved by the proposed framework. The low performance of vanilla Lraspp is attributed to a sever overfitting to the training data, hence, it negatively affected generalisability capabilities on unseen test data. Adding Hue variations during training lessen the overfitting suffered by vanilla Lraspp, hence, it performed much better compared without the usage of Hue component. Furthermore, it is worth noting that Lraspp with Hue component achieved analogous results to Lraspp with only the proposed $TV_{\emptyset}$ unit. Adding texture variations along with Hue variations acquired Lraspp model colour and texture invariance properties and lessen overfitting problem. Accordingly, Lraspp achieved a better on unseen test set which signifies better generalisability capabilities.

Table 27 Test results of the Kvasir-SEG dataset. Models were trained and validated using CVC-ClinicDB dataset.

Kvasir-SEG Test set						
Model	**Acc**	**IoU**	**Dice**	**Rec**	**Prec**	**mIOU**
Lraspp	85.35	36.43	48.64	53.95	55.44	60.19
Lraspp + Hue $H(\hat{x}\|x)$	88.75	44.2	55.3	52.54	73.69	66.03
Lraspp + total variational $TV_{\emptyset}(\check{x}\|x)$	88.04	45.14	57.36	**59.57**	67.07	66.02
Lraspp + $H(\hat{x}\|x)$ + $TV_{\emptyset}(\check{x}\|x)$	**89.66**	**50.69**	**62.04**	57.9	**79.22**	**69.69**
Lraspp + $H(\hat{x}\|x)$ + $TI_{\emptyset}(\bar{x}\|x)$	**89.84**	**50.99**	**62.22**	**59.61**	**77.86**	**69.96**

Comparative results on Kvasir-SEG are noticed for both $TI_{\emptyset}$ and $TV_{\emptyset}$. However, Lraspp segmentation model achieved its highest results only when both Hue and $TI_{\emptyset}$ components were used, which indicates the gain of minimizing the nested accumulated negative log-likelihood $-\mathbb{E}_{P_{data(x)}}\left[logP_{\theta}(y|x) + \mathbb{E}_{H(\hat{x}|x)}logP_{\theta}(y|\hat{x}) + \mathbb{E}_{TI_{\emptyset}(\bar{x}|x)}logP_{\theta}(y|\bar{x})\right]$ using stochastic gradient descent.

Qualitative gradient analysis

To deeply understand the transformations imposed by the proposed $TV_{\emptyset}$ and $TI_{\emptyset}$ units, image's gradient was applied to a training image sample as well as its transformed version, as shown in Figure 42, Figure 43, and Figure 44.

Notice that $TV_{\emptyset}$ completely distorted background textures and edges, meanwhile, it left polyp structure intact. This observation is evident given the gradient of the $TV_{\emptyset}$ image. Furthermore, it is noticed that specular reflections on the background of the $TV_{\emptyset}$ transformed image are completely decayed and disappeared. Exceptions are noticed when those specular reflections are on the polyp itself, as seen in Figure 42, Figure 43, and Figure 44. This behaviour is expected due to the dictated objective which entitled the $TV_{\emptyset}$ unit to preserve polyp texture regardless of the presence of specularity or other remaining residuals. Moreover, due to black borders surrounding the images, the highest gradient values are found around borders' edges, though, the transformed $TV_{\emptyset}$ images had successfully cleared those artificial edges.

On the other hand, inspecting the gradient of $TI_{\emptyset}$ images indicates that the applied transformation smoothed only mucosa texture, though, it preserved the main edges, as seen in Figure 42, Figure 43, and Figure 44. This is due to the training objectives imposed to the autoencoder as discussed in section 4.2.3. The autoencoder that generate textureless images has a loss consisting of two terms, in which the first term is a reconstruction loss $\|x - \tilde{x}\|$ and the second term to prevent the autoencoder from learning identity function $BCE(y, \tilde{y})$. Accordingly, the autoencoder learned to generate images that have main features found in the corresponding original images.

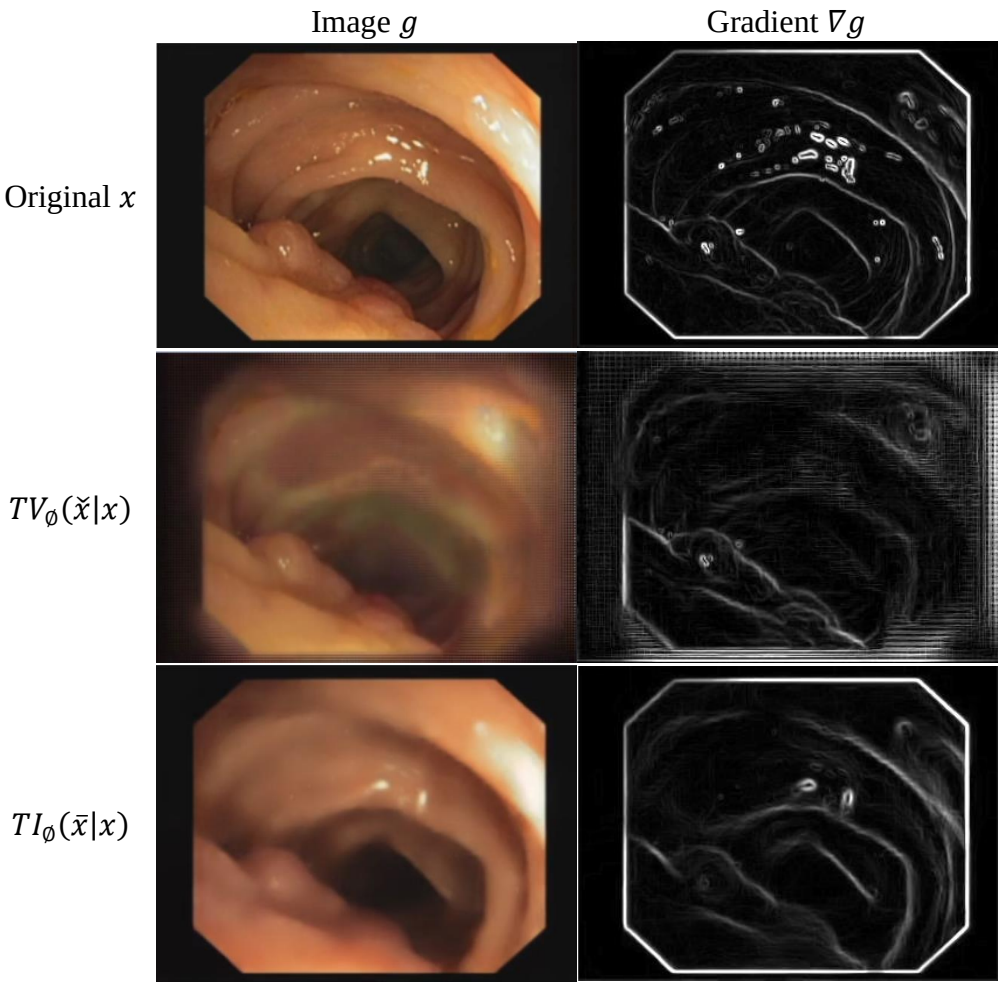


Figure 42 Gradient analyses of a transformed image using $TV_{\emptyset}$ and $TI_{\emptyset}$, respectively.

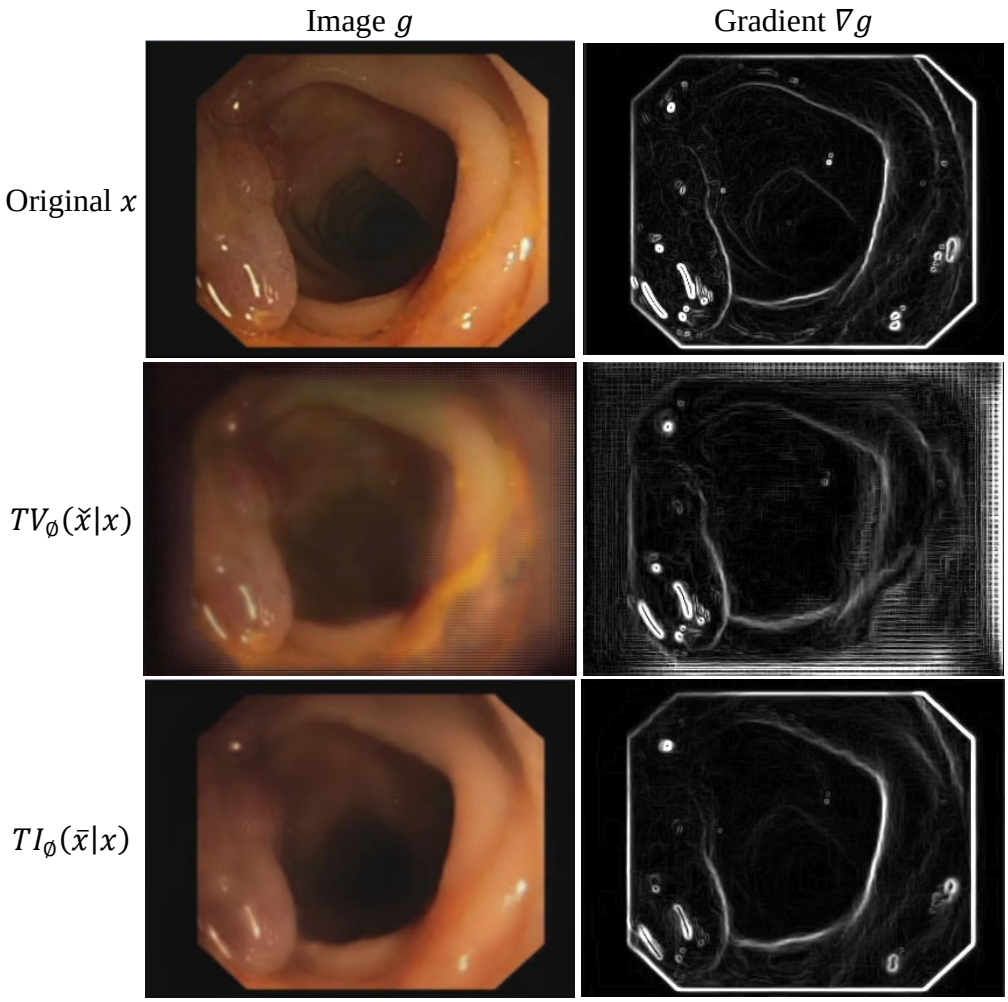


Figure 43 Gradient analyses of another transformed image using $TV_{\emptyset}$ and $TI_{\emptyset}$, respectively.

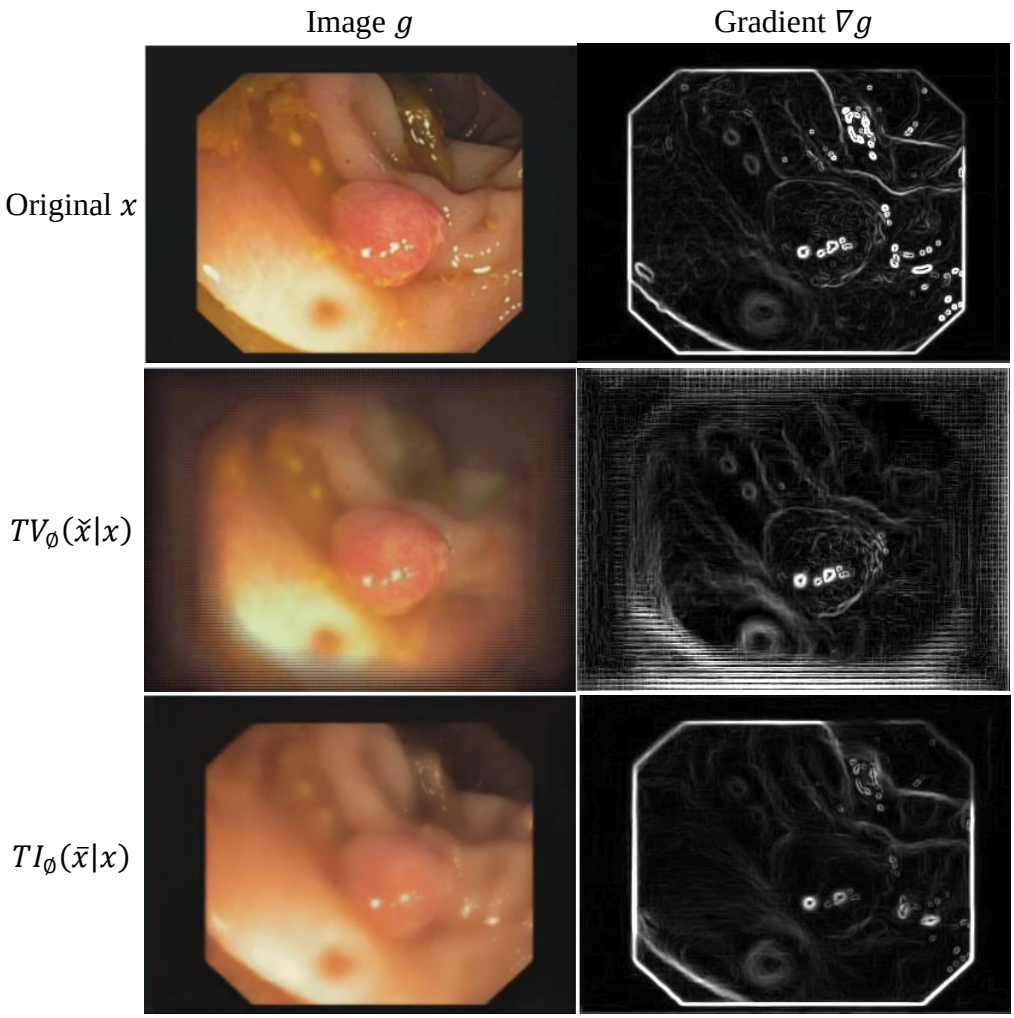


Figure 44 Gradient analyses of another transformed image using $TV_{\emptyset}$ and $TI_{\emptyset}$, respectively.

Furthermore, the effect of $TV_{\emptyset}$ transformation can be seen as an anomaly pattern detection unit in which polyp patterns are considered to be a valid pattern. Meanwhile any other patterns that deviate significantly from polyp patterns are considered an outlier, hence, it got washed away during the transformation, as seen in Figure 45. It is worth noting that during the training phase CVC-ClinicDB was utilised which does not have any image that contains instrument, nevertheless, the $TV_{\emptyset}$ washed away an instrument region of an unseen test image, as seen in Figure 45. Since $TV_{\emptyset}$ unit was not exposed to such pattern, it suggests that the $TV_{\emptyset}$ unit considered the instrument region as an outlier, hence, the gradients of that regions were reduced.

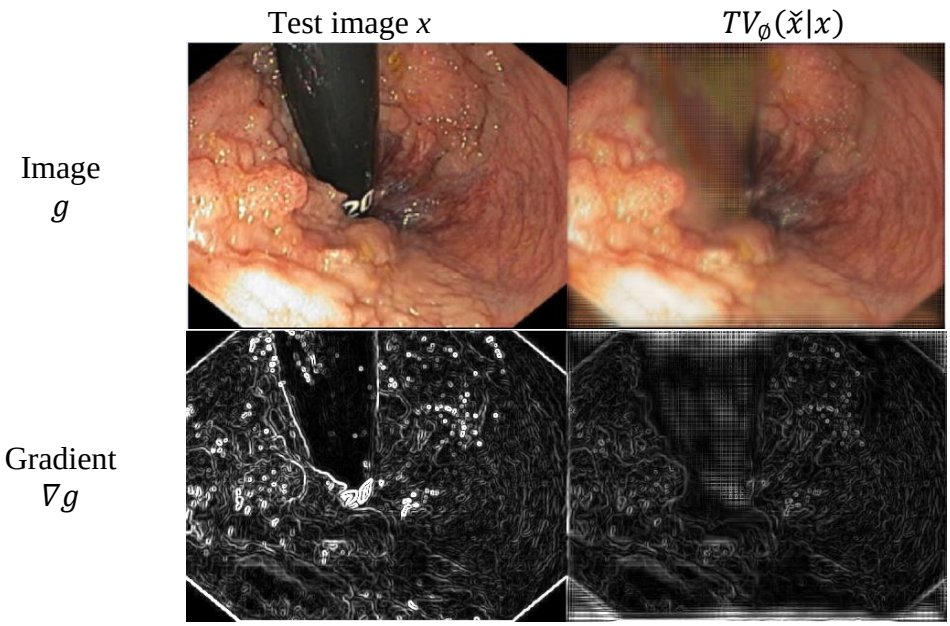


Figure 45 The effect of $TV_{\emptyset}$ transformation can be seen as an anomaly-detection unit.

4.4 Summary

This chapter introduced a novel framework aimed at improving the generalisability of segmentation models. The limited availability of polyp datasets poses a challenge to deep learning-based segmentation models. While typical methods implicate data augmentation or generative models to inflate the training datasets, these approaches have not shown significant effectiveness in the literature. Contrary to the current methods, our proposed approach suggests employing image-to-image transformations in conjunction with a segmentation model to enhance its performance. The proposed framework consists of texture-based unit and random hue shifting unit to variate texture and colour of input images, respectively. Two concrete implementations of the texture unit are proposed, including, Total Variational $TV_{\emptyset}$ and Texture Interpolation $TI_{\emptyset}$. Experimental results have shown that the proposed framework consistently enhances segmentation results. The experimental results show that the proposed framework achieved improvements in Intersection over Union (IoU) ranging from approximately 1.8% to 16.4% across three different test sets when considering only the polyp mask. However, when considering the background mask in addition to the polyp mask, the improvements in mean Intersection over Union (mIoU) were even more significant. The mIoU improvements ranged from approximately 1.2% to 21.9% across the unseen test sets. These results indicate that the proposed framework not only enhances the segmentation accuracy of polyps but also improves the overall segmentation performance, including the accurate delineation of the background region. The reported enhancements highlight the effectiveness of the proposed framework in enhancing the generalisability of segmentation models, irrespective of their internal architectural design.

Chapter 5

Conclusions and future work

Within this chapter, the entirety of our thesis conclusions is encapsulated. To this end, the chapter is subdivided into two distinct sections, each offering a platform for showcasing various dimensions of this endeavour. In the initial section , conclusions will be drawn from both Chapter 3 and Chapter 4, which encapsulate our efforts in addressing certain voids identified within the existing literature. Subsequently, an exploration of our vision for future work and the subsequent trajectory of our research is presented.

5.1 Contributions

Given the recent remarkable achievements of deep learning in diverse computer vision tasks, there has been a natural assumption that this success would seamlessly extend to other domains, including colonoscopy. However, this thesis reveals that the success achieved in non-medical domains does not automatically translate to the domain of colonoscopy. This conclusion, while it may appear straightforward, is actually quite nuanced, especially in light of the consistently high results reported in the literature regarding the application of deep learning in colonoscopy. Consequently, the application of deep learning in colonoscopy might be prematurely perceived as ready for use in real clinical practice, whereas the reality is that these models are still in the preclinical stage.

To illustrate this unfortunate reality, two distinct yet interconnected tasks within colonoscopy were chosen, namely, bowel preparation assessment and polyp segmentation. Both tasks were addressed using deep learning in Chapter 3 and Chapter 4, respectively. In the first task, the colon undergoes an extensive cleansing process to eliminate any stool or residues that might obscure the visibility of polyps during screening. After bowel preparation, a colonoscopist performs screening to detect any abnormal inflammation or mucosal growth (i.e., polyps) and removes them accordingly. Subsequently, the colonoscopist compiles a report summarizing the findings.

Contributions of this thesis are listed as follows:

- In this thesis, two distinct deep learning models were introduced and designed to provide support to colonoscopists. The first model serves to automatically evaluate the degree of bowel preparation, thereby optimizing time and energy allocation for other tasks (as discussed in Chapter 3). The second model is focused on the automatic identification of polyp locations and the subsequent generation of polyp masks (as discussed in Chapter 4). This automatic polyp segmentation process significantly enhances the accuracy of polyp screening, ultimately

reducing the likelihood of missed polyps during the screening procedure.

- For both tasks (i.e., automatic bowel preparation and polyp segmentation), subtle overestimations results in the literature had been elaborated in the thesis. Colonoscopy datasets consist of semi-consecutive frames or sometimes videos. Hence, arbitrary dividing such datasets result in subsets that closely resemble each other, consequently leading to inflated performance metrics during validation and testing. To substantiate this assert, experiments were conducted in this study that show the gap between real and inflated results.

- A novel video-based deep learning model is proposed in Chapter 3. We argue that using video-based deep learning models to evaluate bowel cleansing degree is better than frame-based model due to contextual information embedded in videos. Accordingly, a video-based model was designed to classify bowel preparation degree on video samples. The proposed model leverages both sequential information and spatial information by utilising recurrent neural network unit (i.e., GRU) and a proposed Multiplexer unit, respectively. Consequently, mathematical analysis was conducted to justify the chosen normalisation method. Finally, the proposed model was evaluated against frame-based deep learning models as well as video-based deep learning models. Experiment results consistently indicated the superiority of the proposed model against state-of-the-art models.

- A novel segmentation framework is proposed in Chapter 4. The proposed deep learning framework enhances the generalisability by utilising two deep learning components, namely, transformation unit and a segmentation model. The segmentation model gets as input out-of-domain images with correct corresponding masks using the image-to-image transformation unit. Both the transformation unit and the segmentation model are two separate deep learning models. The fundamental concept behind the proposed deep learning framework is to instil colour and texture detail invariance properties into any segmentation model, thus enhancing its generalisability.

- A mathematical justification was presented in Chapter 4 for the proposed segmentation framework. The learning phase of the proposed framework can be seen as performing gradient-based approximate minimization on nested negative log-likelihood. This is due to the segmentation loss imposed for each training image as well as a corresponding series of transformed versions.

- Two concrete implementations of the proposed segmentation framework were presented in Chapter 4, including minimizing total variations $TV_{\emptyset}$ and texture interpolation $TI_{\emptyset}$. The proposed $TV_{\emptyset}$ unit can be described as an autoencoder model featuring a unique loss function.

The loss function computes the overall image gradients, known as Total Variation, for the background. The primary aim of the $TV_{\emptyset}$ unit is to minimize these gradients within the less critical regions of the images, notably the background, all the while maintaining the integrity of the regions of interest, such as the polyps. On the other hand, the proposed $TI_{\emptyset}$ employ spatial interpolation between input images and corresponding textureless version. The textureless images are produced by a pretrained autoencoder.

- Extensive experimentations were conducted to showcase the effectiveness of the proposed framework in improving generalisability and performance on unseen test datasets. The proposed framework is tested against conventional segmentation models and the state-of-the-art models with/without augmentations. Furthermore, the proposed framework was evaluated using a published benchmark (i.e., EndoSceneStill benchmark). In terms of polyp Intersection over Union (IoU) on the unseen test set, the proposed framework outperformed the reported benchmark by approximately 12%. The results of these experiments clearly demonstrate that the proposed framework significantly enhanced the generalisability of the used segmentation model regardless of its internal architectural design.

5.2 Future work

The task of automatic bowel preparation assessment is generally regarded as less complex than polyp segmentation. This is primarily due to the fact that polyps lack a distinct and specific shape, making it challenging for both humans and deep learning models to accurately delineate the precise boundaries of these abnormal growths. On the other hand, in the case of automatic bowel preparation assessment, there exist similarities in the discrete clarity levels of the bowel, which can introduce ambiguity for both humans and models in determining the actual class for certain instances. Nevertheless, it is worth noting that the performance of deep learning models in both tasks can be improved.

Currently, the problem of automatic bowel preparation evaluation is treated as classification task in which frames or videos belong to one of four discrete classes accordingly to the clarity level. Instead, this problem could be treated as a regression task in which frames or videos should be given a number between [0,1] to indicate the clarity level. The nature of this problem may need such uncertainty treatment to further enhance performance of deep learning models and providing justifiable results. Accordingly, better clinical reports would be expected hence encourage the usage of deep learning models in real-clinical environments.

A big high quality polyp dataset is not expected in the near future; however, we expect to have various unlabelled colonoscopy datasets. Such unlabelled data are currently utilised using unsupervised learning methods to enhance feature representations produced by deep learning models. Instead, this unlabelled data could be utilised in a novel way to facilitate generating out-of-domain labelled images. For instance, unlabelled data could facilitate learning training manifold and accordingly one can utilise that to generate new images on-the-fly to exposed segmentation models to new pattern hence enhance the generalisability of segmentation models. Furthermore, prior knowledge about polyp topology should be considered to extend accuracy of segmentation models specially on unseen samples.

References

[1] "Digestive Diseases Statistics for the United States | NIDDK," National Inistitution of Diabetes and Digestive and Kidney Diseases. Accessed: Feb. 12, 2020. [Online]. Available: https://www.niddk.nih.gov/health-information/health-statistics/digestive-diseases#all

[2] K. Pogorelov, K. R. Randel, C. Griwodz, S. L. Eskeland, T. De Lange, D. Johansen, C. Spampinato, D. T. Dang-Nguyen, M. Lux, P. T. Schmidt, M. Riegler, and P. Halvorsen, "Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection," in *Proceedings of the 8th ACM Multimedia Systems Conference, MMSys 2017*, 2017. doi: 10.1145/3083187.3083212.

[3] "Cancer: fact sheets," World Health Organization (WHO). Accessed: Jul. 31, 2023. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/cancer

[4] "Gastrointestinal tract - Mayo Clinic." Accessed: Feb. 08, 2021. [Online]. Available: https://www.mayoclinic.org/gastrointestinal-tract/img-20007468

[5] Y. S. He, J. R. Su, Z. Li, X. L. Zuo, and Y. Q. Li, "Application of artificial intelligence in gastrointestinal endoscopy," *J. Dig. Dis.*, pp. 1751-2980.12827, Nov. 2019, doi: 10.1111/1751-2980.12827.

[6] J. de Groof, F. van der Sommen, J. van der Putten, M. R. Struyvenberg, S. Zinger, W. L. Curvers, O. Pech, A. Meining, H. Neuhaus, R. Bisschops, E. J. Schoon, P. H. de With, and J. J. Bergman, "The Argos project: The development of a computer-aided detection system to improve detection of Barrett's neoplasia on white light endoscopy," *United Eur. Gastroenterol. J.*, vol. 7, no. 4, pp. 538–547, 2019, doi: 10.1177/2050640619837443.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, pp. 234–241. doi: 10.1007/978-3-319-24574-4_28.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[10] A. Ebigbo, C. Palm, A. Probst, R. Mendel, J. Manzeneder, F. Prinz, L. A. de Souza, J. P. Papa, P. Siersema, and H. Messmann, "A technical

review of artificial intelligence as applied to gastrointestinal endoscopy: clarifying the terminology," *Endosc. Int. Open*, vol. 07, no. 12, pp. E1616–E1623, Dec. 2019, doi: 10.1055/a-1010-5705.

[11] A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *New England Journal of Medicine*, vol. 380, no. 14. pp. 1347–1358, 2019. doi: 10.1056/NEJMra1814259.

[12] A. Asperti and C. Mastronardo, "The effectiveness of data augmentation for detection of gastrointestinal diseases from endoscopical images," in *BIOIMAGING 2018 - 5th International Conference on Bioimaging, Proceedings; Part of 11th International Joint Conference on Biomedical Engineering Systems and Technologies, BIOSTEC 2018*, 2018. doi: 10.5220/0006730901990205.

[13] O. F. Ahmad, D. Stoyanov, and L. B. Lovat, "Barriers and pitfalls for artificial intelligence in gastroenterology: Ethical and regulatory issues," *Tech. Gastrointest. Endosc.*, p. 150636, Oct. 2019, doi: 10.1016/J.TGIE.2019.150636.

[14] V. L. Thambawita, I. Strümke, S. Hicks, M. A. Riegler, P. Halvorsen, and S. Parasa, "ID: 3523524 DATA AUGMENTATION USING GENERATIVE ADVERSARIAL NETWORKS FOR CREATING REALISTIC ARTIFICIAL COLON POLYP IMAGES: VALIDATION STUDY BY ENDOSCOPISTS," *Gastrointest. Endosc.*, vol. 93, no. 6, p. AB190, Jun. 2021, doi: 10.1016/j.gie.2021.03.431.

[15] Y. Shin, H. A. Qadir, and I. Balasingham, "Abnormal Colon Polyp Image Synthesis Using Conditional Adversarial Networks for Improved Detection Performance," *IEEE Access*, vol. 6, pp. 56007–56017, 2018, doi: 10.1109/ACCESS.2018.2872717.

[16] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[17] "Definition of endoscopy - NCI Dictionary of Cancer Terms - National Cancer Institute." Accessed: Feb. 08, 2021. [Online]. Available: https://www.cancer.gov/publications/dictionaries/cancer-terms/def/endoscopy

[18] B. Münzer, K. Schoeffmann, and L. Böszörmenyi, "Content-based processing and analysis of endoscopic images and videos: A survey," *Multimed. Tools Appl.*, 2018, doi: 10.1007/s11042-016-4219-z.

[19] W. Y. Cho, J. Y. Jang, and D. H. Lee, "Recent advances in image-enhanced endoscopy," *Clinical Endoscopy*. 2011. doi: 10.5946/ce.2011.44.2.65.

[20] M. Song and T. L. Ang, "Early detection of early gastric cancer using image-enhanced endoscopy: Current trends," *Gastrointest. Interv.*, 2014, doi: 10.1016/j.gii.2014.02.005.

[21] K. Yao, "The endoscopic diagnosis of early gastric cancer," *Annals of Gastroenterology*. 2013.

[22] "Types of Endoscopy | Cancer.Net." Accessed: Feb. 09, 2021. [Online]. Available: https://www.cancer.net/navigating-cancer-care/diagnosing-cancer/tests-and-procedures/types-endoscopy

[23] V. X. Nguyen, T. Le Nguyen, C. C. Nguyen, and X. Nguyen, "Appropriate use of endoscopy in the diagnosis and treatment of gastrointestinal diseases: up-to-date indications for primary care providers," *Int. J. Gen. Med.*, vol. 2010, pp. 3–345, 2010, doi: 10.2147/IJGM.S14555.

[24] "clevelandclinic: Colonoscopy." Accessed: Aug. 09, 2023. [Online]. Available: https://my.clevelandclinic.org/health/diagnostics/4949-colonoscopy

[25] D. S. Early, T. Ben-Menachem, G. A. Decker, J. A. Evans, R. D. Fanelli, D. A. Fisher, N. Fukami, J. H. Hwang, R. Jain, T. L. Jue, K. M. Khan, P. M. Malpas, J. T. Maple, R. S. Sharaf, J. A. Dominitz, and B. D. Cash, "Appropriate use of GI endoscopy," *Gastrointest. Endosc.*, vol. 75, no. 6, pp. 1127–1131, 2012, doi: 10.1016/j.gie.2012.01.011.

[26] Y. S. He, J. R. Su, Z. Li, X. L. Zuo, and Y. Q. Li, "Application of artificial intelligence in gastrointestinal endoscopy," *J. Dig. Dis.*, vol. 20, no. 12, pp. 623–630, 2019, doi: 10.1111/1751-2980.12827.

[27] P. Wang, T. M. Berzin, J. R. Glissen Brown, S. Bharadwaj, A. Becq, X. Xiao, P. Liu, L. Li, Y. Song, D. Zhang, Y. Li, G. Xu, M. Tu, and X. Liu, "Real-time automatic detection system increases colonoscopic polyp and adenoma detection rates: A prospective randomised controlled study," *Gut*, vol. 68, no. 10, pp. 1813–1819, Oct. 2019, doi: 10.1136/gutjnl-2018-317500.

[28] A. Qayyum, J. Qadir, M. Bilal, and A. Al-Fuqaha, "Secure and Robust Machine Learning for Healthcare: A Survey," *IEEE Rev. Biomed. Eng.*, vol. 14, pp. 156–180, 2021, doi: 10.1109/RBME.2020.3013489.

[29] W. Du, N. Rao, D. Liu, H. Jiang, C. Luo, Z. Li, T. Gan, and B. Zeng, "Review on the Applications of Deep Learning in the Analysis of Gastrointestinal Endoscopy Images," *IEEE Access*, vol. 7, pp. 142053–142069, 2019, doi: 10.1109/access.2019.2944676.

[30] T. Boers, J. van der Putten, M. Struyvenberg, K. Fockens, J. Jukema, E. Schoon, F. van der Sommen, J. Bergman, and P. de With, "Improving temporal stability and accuracy for endoscopic video tissue classification using recurrent neural networks," *Sensors (Switzerland)*, vol. 20, no. 15, pp. 1–11, 2020, doi: 10.3390/s20154133.

[31] M. S. Haithami, A. Ahmed, I. Y. Liao, and H. J. Altulea, "Automatic Bowel Preparation Assessment Using Deep Learning," in *Pattern Recognition, Computer Vision, and Image Processing. ICPR 2022 International Workshops and Challenges*, J.-J. Rousseau and B. Kapralos, Eds., Springer Nature Switzerland, 2023, pp. 574–588. doi: 10.1007/978-3-031-37660-3_40.

[32] S. Ali, N. Ghatwary, D. Jha, E. Isik-Polat, G. Polat, C. Yang, W. Li, A.

Galdran, M.-Á. G. Ballester, V. Thambawita, S. Hicks, S. Poudel, S.-W. Lee, Z. Jin, T. Gan, C. Yu, J. Yan, D. Yeo, H. Lee, N. K. Tomar, M. Haithami, A. Ahmed, M. A. Riegler, C. Daul, P. Halvorsen, J. Rittscher, O. E. Salem, D. Lamarque, R. Cannizzaro, S. Realdon, T. de Lange, and J. E. East, "Assessing generalisability of deep learning-based polyp detection and segmentation methods through a computer vision challenge," *Sci. Rep.*, vol. 14, no. 1, p. 2032, Jan. 2024, doi: 10.1038/s41598-024-52063-x.

[33] A. Lemay, K. Hoebel, C. P. Bridge, B. Befano, S. De Sanjosé, D. Egemen, A. C. Rodriguez, M. Schiffman, J. P. Campbell, and J. Kalpathy-Cramer, "Improving the repeatability of deep learning models with Monte Carlo dropout," *npj Digit. Med.*, vol. 5, no. 1, p. 174, Nov. 2022, doi: 10.1038/s41746-022-00709-3.

[34] M. Crane, "Questionable Answers in Question Answering Research: Reproducibility and Variability of Published Results," *Trans. Assoc. Comput. Linguist.*, vol. 6, pp. 241–252, Dec. 2018, doi: 10.1162/tacl_a_00018.

[35] S. Hicks, D. Jha, V. Thambawita, P. Halvorsen, B.-J. Singstad, S. Gaur, K. Pettersen, M. Goodwin, S. Parasa, T. De Lange, and M. Riegler, "MedAI: Transparency in Medical Image Segmentation," *Nord. Mach. Intell.*, vol. 1, no. 1, pp. 1–4, Nov. 2021, doi: 10.5617/nmi.9140.

[36] J. Zhou, L. Wu, X. Wan, L. Shen, J. Liu, J. Zhang, X. Jiang, Z. Wang, S. Yu, J. Kang, M. Li, S. Hu, X. Hu, D. Gong, D. Chen, L. Yao, Y. Zhu, and H. Yu, "A novel artificial intelligence system for the assessment of bowel preparation (with video)," *Gastrointest. Endosc.*, vol. 91, no. 2, pp. 428-435.e2, Feb. 2020, doi: 10.1016/j.gie.2019.11.026.

[37] S. Hwang, J. H. Oh, W. Tavanapong, J. Wong, and P. C. De Groen, "Stool detection in colonoscopy videos," in *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS'08 - "Personalized Healthcare through Technology,"* 2008. doi: 10.1109/iembs.2008.4649835.

[38] J. Muthukudage, J. Oh, W. Tavanapong, J. Wong, and P. C. De Groen, "Color based stool region detection in colonoscopy videos for quality measurements," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7087 LNCS, no. PART1, pp. 61–72, 2011, doi: 10.1007/978-3-642-25367-6_6.

[39] K. Pogorelov, K. R. Randel, T. De Lange, S. L. Eskeland, C. Griwodz, D. Johansen, C. Spampinato, M. Taschwer, M. Lux, P. T. Schmidt, M. Riegler, and P. Halvorsen, "Nerthus: A bowel preparation quality video dataset," in *Proceedings of the 8th ACM Multimedia Systems Conference, MMSys 2017*, New York, NY, USA: ACM, Jun. 2017, pp. 170–174. doi: 10.1145/3083187.3083216.

[40] L. Guo, X. Xiao, C. Wu, X. Zeng, Y. Zhang, J. Du, S. Bai, J. Xie, Z. Zhang, Y. Li, X. Wang, O. Cheung, M. Sharma, J. Liu, and B. Hu, "Real-time automated diagnosis of precancerous lesions and early esophageal

squamous cell carcinoma using a deep learning model (with videos)," *Gastrointest. Endosc.*, vol. 91, no. 1, pp. 41–51, Jan. 2020, doi: 10.1016/j.gie.2019.08.018.

[41] A. J. de Groof, M. R. Struyvenberg, K. N. Fockens, J. van der Putten, F. van der Sommen, T. G. Boers, S. Zinger, R. Bisschops, P. H. de With, R. E. Pouw, W. L. Curvers, E. J. Schoon, and J. J. G. H. M. Bergman, "Deep learning algorithm detection of Barrett's neoplasia with high accuracy during live endoscopic procedures: a pilot study (with video)," *Gastrointest. Endosc.*, vol. 91, no. 6, pp. 1242–1250, 2020, doi: 10.1016/j.gie.2019.12.048.

[42] N. Ghatwary, M. Zolgharni, F. Janan, and X. Ye, "Learning Spatiotemporal Features for Esophageal Abnormality Detection From Endoscopic Videos," *IEEE J. Biomed. Heal. Informatics*, vol. 25, no. 1, pp. 131–142, Jan. 2021, doi: 10.1109/JBHI.2020.2995193.

[43] M. Hussein, J. Gonzalez-Bueno Puyal, P. Brandao, D. Toth, V. Sehgal, M. A. Everson, G. Lipman, O. F. Ahmad, R. Kader, J. M. Esteban, R. Bisschops, M. Banks, P. Mountney, D. Stoyanov, L. Lovat, and R. Haidry, "DEEP NEURAL NETWORK FOR THE DETECTION OF EARLY NEOPLASIA IN BARRETT'S OESOPHAGUS," *Gastrointest. Endosc.*, vol. 91, no. 6, p. AB250, Jun. 2020, doi: 10.1016/j.gie.2020.03.1826.

[44] H. Nakashima, "Artificial intelligence diagnosis of Helicobacter pylori infection using blue laser imaging-bright and linked color imaging: a single-center prospective study," *Ann. Gastroenterol.*, 2018, doi: 10.20524/aog.2018.0269.

[45] Y. Zhang, F. Li, F. Yuan, K. Zhang, L. Huo, Z. Dong, Y. Lang, Y. Zhang, M. Wang, Z. Gao, Z. Qin, and L. Shen, "Diagnosing chronic atrophic gastritis by gastroscopy using artificial intelligence," *Dig. Liver Dis.*, vol. 52, no. 5, pp. 566–572, May 2020, doi: 10.1016/j.dld.2019.12.146.

[46] T. Yan, P. K. Wong, I. C. Choi, C. M. Vong, and H. H. Yu, "Intelligent diagnosis of gastric intestinal metaplasia based on convolutional neural network and limited number of endoscopic images," *Comput. Biol. Med.*, vol. 126, p. 104026, Nov. 2020, doi: 10.1016/j.compbiomed.2020.104026.

[47] T. K. L. Lui, K. K. Y. Wong, L. L. Y. Mak, E. W. P. To, V. W. M. Tsui, Z. Deng, J. Guo, L. Ni, M. K. S. Cheung, and W. K. Leung, "Feedback from artificial intelligence improved the learning of junior endoscopists on histology prediction of gastric lesions," *Endosc. Int. Open*, vol. 08, no. 02, pp. E139–E146, Feb. 2020, doi: 10.1055/a-1036-6114.

[48] T. Yan, P. K. Wong, and Y.-Y. Qin, "Deep learning for diagnosis of precancerous lesions in upper gastrointestinal endoscopy: A review," *World J. Gastroenterol.*, vol. 27, no. 20, pp. 2531–2544, May 2021, doi: 10.3748/wjg.v27.i20.2531.

[49] A. Ebigbo, R. Mendel, A. Probst, J. Manzeneder, L. A. De Souza, J. P. Papa, C. Palm, and H. Messmann, "Computer-aided diagnosis using deep

learning in the evaluation of early oesophageal adenocarcinoma," *Gut*, vol. 68, no. 7. BMJ Publishing Group, pp. 1143–1145, Jul. 01, 2019. doi: 10.1136/gutjnl-2018-317573.

[50] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems*, 2015.

[51] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with Gaussian edge potentials," in *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, 2011.

[52] S. Shichijo, S. Nomura, K. Aoyama, Y. Nishikawa, M. Miura, T. Shinagawa, H. Takiyama, T. Tanimoto, S. Ishihara, K. Matsuo, and T. Tada, "Application of Convolutional Neural Networks in the Diagnosis of Helicobacter pylori Infection Based on Endoscopic Images," *EBioMedicine*, vol. 25, pp. 106–111, Nov. 2017, doi: 10.1016/j.ebiom.2017.10.014.

[53] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2015, pp. 1–9. doi: 10.1109/CVPR.2015.7298594.

[54] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017. doi: 10.1109/CVPR.2017.243.

[55] J. Arribas, G. Antonelli, L. Frazzoni, L. Fuccio, A. Ebigbo, F. Van Der Sommen, N. Ghatwary, C. Palm, M. Coimbra, F. Renna, J. J. G. H. M. Bergman, P. Sharma, H. Messmann, C. Hassan, and M. J. Dinis-Ribeiro, "Standalone performance of artificial intelligence for upper GI neoplasia: A meta-analysis," *Gut*, pp. 1–11, 2020, doi: 10.1136/gutjnl-2020-321922.

[56] L. F. Sánchez-Peralta, L. Bote-Curiel, A. Picón, F. M. Sánchez-Margallo, and J. B. Pagador, "Deep learning to find colorectal polyps in colonoscopy: A systematic literature review," *Artificial Intelligence in Medicine*, vol. 108. Elsevier B.V., p. 101923, Aug. 01, 2020. doi: 10.1016/j.artmed.2020.101923.

[57] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, "Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement," *PLoS Med.*, vol. 6, no. 7, p. e1000097, Jul. 2009, doi: 10.1371/journal.pmed.1000097.

[58] M. Misawa, S. Kudo, Y. Mori, Y. Maeda, Y. Ogawa, K. Ichimasa, T. Kudo, K. Wakamura, T. Hayashi, H. Miyachi, T. Baba, F. Ishida, H. Itoh, M. Oda, and K. Mori, "Current status and future perspective on artificial intelligence for lower endoscopy," *Dig. Endosc.*, vol. 33, no. 2, pp. 273–

284, Jan. 2021, doi: 10.1111/den.13847.

[59] A. Nogueira-Rodríguez, R. Domínguez-Carbajales, H. López-Fernández, Á. Iglesias, J. Cubiella, F. Fdez-Riverola, M. Reboiro-Jato, and D. Glez-Peña, "Deep Neural Networks approaches for detecting and classifying colorectal polyps," *Neurocomputing*, vol. 423, pp. 721–734, 2021, doi: 10.1016/j.neucom.2020.02.123.

[60] N. M. Mansour, "Artificial Intelligence in Colonoscopy," *Curr. Gastroenterol. Rep.*, vol. 25, no. 6, pp. 122–129, 2023, doi: 10.1007/s11894-023-00872-x.

[61] M. Abdelrahim, H. Saiga, N. Maeda, E. Hossain, H. Ikeda, and P. Bhandari, "Automated sizing of colorectal polyps using computer vision," *Gut*, vol. 71, no. 1, pp. 7–9, Jan. 2022, doi: 10.1136/gutjnl-2021-324510.

[62] I. Barua, M. Misawa, J. R. Glissen Brown, T. Walradt, S. Kudo, S. G. Sheth, J. Nee, J. Iturrino, R. Mukherjee, C. P. Cheney, M. S. Sawhney, D. K. Pleskow, K. Mori, M. Løberg, M. Kalager, P. Wieszczy, M. Bretthauer, T. M. Berzin, and Y. Mori, "Speedometer for withdrawal time monitoring during colonoscopy: a clinical implementation trial," *Scand. J. Gastroenterol.*, vol. 58, no. 6, pp. 664–670, Jun. 2023, doi: 10.1080/00365521.2022.2154616.

[63] E. Klang, Y. Barash, R. Y. Margalit, S. Soffer, O. Shimon, A. Albshesh, S. Ben-Horin, M. M. Amitai, R. Eliakim, and U. Kopylov, "Deep learning algorithms for automated detection of Crohn's disease ulcers by video capsule endoscopy," *Gastrointest. Endosc.*, vol. 91, no. 3, pp. 606-613.e2, Mar. 2020, doi: 10.1016/j.gie.2019.11.012.

[64] X. Luo, J. Zhang, Z. Li, and R. Yang, "Diagnosis of ulcerative colitis from endoscopic images based on deep learning," *Biomed. Signal Process. Control*, vol. 73, p. 103443, Mar. 2022, doi: 10.1016/j.bspc.2021.103443.

[65] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jul. 2017, pp. 1800–1807. doi: 10.1109/CVPR.2017.195.

[66] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Two-Stream Deep Feature Modelling for Automated Video Endoscopy Data Analysis," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12263 LNCS, 2020, pp. 742–751. doi: 10.1007/978-3-030-59716-0_71.

[67] Y. Zhu, Y. Xu, W. Chen, T. Zhao, and S. Zheng, "A CNN-based Cleanliness Evaluation for Bowel Preparation in Colonoscopy," *Proc. - 2019 12th Int. Congr. Image Signal Process. Biomed. Eng. Informatics, CISP-BMEI 2019*, pp. 1–5, 2019, doi: 10.1109/CISP-BMEI48845.2019.8965825.

[68] J. Amin, M. Sharif, E. Gul, and R. S. Nayak, “3D-semantic segmentation and classification of stomach infections using uncertainty aware deep neural networks,” *Complex Intell. Syst.*, no. 0123456789, 2021, doi: 10.1007/s40747-021-00328-7.

[69] M. Riegler, K. Pogorelov, P. Halvorsen, T. De Lange, C. Griwodz, P. T. Schmidt, S. L. Eskeland, and D. Johansen, “EIR - Efficient computer aided diagnosis framework for gastrointestinal endoscopies,” in *Proceedings - International Workshop on Content-Based Multimedia Indexing*, 2016. doi: 10.1109/CBMI.2016.7500257.

[70] K. Pogorelov, M. Riegler, S. L. Eskeland, T. de Lange, D. Johansen, C. Griwodz, P. T. Schmidt, and P. Halvorsen, “Efficient disease detection in gastrointestinal videos – global features versus neural networks,” *Multimed. Tools Appl.*, vol. 76, no. 21, pp. 22493–22525, Nov. 2017, doi: 10.1007/s11042-017-4989-y.

[71] A. Santoro, D. Raposo, D. G. T. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap, “A simple neural network module for relational reasoning,” in *Advances in Neural Information Processing Systems*, Neural information processing systems foundation, Jun. 2017, pp. 4968–4977. Accessed: Mar. 29, 2021. [Online]. Available: http://arxiv.org/abs/1706.01427

[72] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[73] A. Krizhevsky, G. Hinton, and others, “Learning multiple layers of features from tiny images,” 2009.

[74] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, “Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries,” *CA. Cancer J. Clin.*, vol. 68, no. 6, pp. 394–424, Nov. 2018, doi: 10.3322/caac.21492.

[75] E. Shelhamer, J. Long, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, 2017, doi: 10.1109/TPAMI.2016.2572683.

[76] M. Akbari, M. Mohrekesh, E. Nasr-Esfahani, S. M. R. Soroushmehr, N. Karimi, S. Samavi, and K. Najarian, “Polyp Segmentation in Colonoscopy Images Using Fully Convolutional Network,” in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, Jul. 2018, pp. 69–72. doi: 10.1109/EMBC.2018.8512197.

[77] P. Brandao, E. Mazomenos, G. Ciuti, R. Caliò, F. Bianchi, A. Menciassi, P. Dario, A. Koulaouzidis, A. Arezzo, and D. Stoyanov, “Fully convolutional neural networks for polyp segmentation in colonoscopy,” in *Medical Imaging 2017: Computer-Aided Diagnosis*, S. G. Armato and N. A. Petrick, Eds., Mar. 2017, p. 101340F. doi: 10.1117/12.2254361.

[78] L. Zhang, S. Dolwani, and X. Ye, “Automated Polyp Segmentation in Colonoscopy Frames Using Fully Convolutional Neural Network and Textons,” in *Annual Conference on Medical Image Understanding and Analysis*, 2017, pp. 707–717. doi: 10.1007/978-3-319-60964-5_62.

[79] G. Yun Bo and M. Bogdan, “Giana polyp segmentation with fully convolutional dilation neural networks,” in *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2019, pp. 632–641.

[80] A. Oulefki, S. Agaian, T. Trongtirakul, and A. Kassah Laouar, “Automatic COVID-19 lung infected region segmentation and measurement using CT-scans images,” *Pattern Recognit.*, vol. 114, p. 107747, Jun. 2021, doi: 10.1016/j.patcog.2020.107747.

[81] D.-P. Fan, T. Zhou, G.-P. Ji, Y. Zhou, G. Chen, H. Fu, J. Shen, and L. Shao, “Inf-Net: Automatic COVID-19 Lung Infection Segmentation From CT Images,” *IEEE Trans. Med. Imaging*, vol. 39, no. 8, pp. 2626–2637, Aug. 2020, doi: 10.1109/TMI.2020.2996645.

[82] W. Li, F. Jia, and Q. Hu, “Automatic Segmentation of Liver Tumor in CT Images with Deep Convolutional Neural Networks,” *J. Comput. Commun.*, vol. 03, no. 11, pp. 146–151, 2015, doi: 10.4236/jcc.2015.311023.

[83] S. U. Akram, J. Kannala, L. Eklund, and J. Heikkilä, “Cell Segmentation Proposal Network for Microscopy Image Analysis,” in *Carneiro, G., et al. Deep Learning and Data Labeling for Medical Applications. DLMIA LABELS 2016 2016. Lecture Notes in Computer Science()*, Springer, 2016, pp. 21–29. doi: 10.1007/978-3-319-46976-8_3.

[84] N.-Q. Nguyen and S.-W. Lee, “Robust Boundary Segmentation in Medical Images Using a Consecutive Deep Encoder-Decoder Network,” *IEEE Access*, vol. 7, pp. 33795–33808, 2019, doi: 10.1109/ACCESS.2019.2904094.

[85] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, “DoubleU-Net: A Deep Convolutional Neural Network for Medical Image Segmentation,” in *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, IEEE, Jul. 2020, pp. 558–564. doi: 10.1109/CBMS49503.2020.00111.

[86] T. Mahmud, B. Paul, and S. A. Fattah, “PolypSegNet: A modified encoder-decoder architecture for automated polyp segmentation from colonoscopy images,” *Comput. Biol. Med.*, vol. 128, p. 104119, Jan. 2021, doi: 10.1016/j.compbiomed.2020.104119.

[87] C.-H. Huang, H.-Y. Wu, and Y.-L. Lin, “HarDNet-MSEG: A Simple Encoder-Decoder Polyp Segmentation Neural Network that Achieves over 0.9 Mean Dice and 86 FPS,” *arXiv*, Jan. 2021, doi: https://doi.org/10.48550/arXiv.2101.07172.

[88] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,”

IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, 2017, doi: 10.1109/TPAMI.2016.2644615.

[89] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv*. 2017.

[90] 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," *Med. Phys.*, vol. 46, no. 12, pp. 5666–5676, Dec. 2019, doi: 10.1002/mp.13865.

[91] X. Jia, X. Mai, Y. Cui, Y. Yuan, X. Xing, H. Seo, L. Xing, and M. Q. H. Meng, "Automatic Polyp Recognition in Colonoscopy Images Using Deep Learning and Two-Stage Pyramidal Feature Prediction," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 3, pp. 1–15, 2020, doi: 10.1109/TASE.2020.2964827.

[92] 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," *Comput. Med. Imaging Graph.*, vol. 43, pp. 99–111, 2015, doi: 10.1016/j.compmedimag.2015.02.007.

[93] X. Sun, P. Zhang, D. Wang, Y. Cao, and B. Liu, "Colorectal polyp segmentation by U-Net with dilation convolution," *Proc. - 18th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2019*, pp. 851–858, Dec. 2019, doi: 10.1109/ICMLA.2019.00148.

[94] S. Safarov and T. K. Whangbo, "A-DenseUNet: Adaptive Densely Connected UNet for Polyp Segmentation in Colonoscopy Images with Atrous Convolution," *Sensors*, vol. 21, no. 4, p. 1441, Feb. 2021, doi: 10.3390/s21041441.

[95] W. T. Xiao, L. J. Chang, and W. M. Liu, "Semantic Segmentation of Colorectal Polyps with DeepLab and LSTM Networks," in *2018 IEEE International Conference on Consumer Electronics-Taiwan, ICCE-TW 2018*, 2018. doi: 10.1109/ICCE-China.2018.8448568.

[96] N.-Q. Nguyen, D. M. Vo, and S.-W. Lee, "Contour-Aware Polyp Segmentation in Colonoscopy Images Using Detailed Upsamling Encoder-Decoder Networks," *IEEE Access*, vol. 8, pp. 99495–99508, 2020, doi: 10.1109/ACCESS.2020.2995630.

[97] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, Neural information processing systems foundation, Jun. 2017, pp. 5999–6009. Accessed: Jun. 20, 2021. [Online]. Available: https://arxiv.org/abs/1706.03762v5

[98] R. Zhang, G. Li, Z. Li, S. Cui, D. Qian, and Y. Yu, "Adaptive Context Selection for Polyp Segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12266 LNCS, 2020, pp. 253–262. doi: 10.1007/978-3-030-59725-2_25.

[99] A. Lou, S. Guan, H. Ko, and M. H. Loew, “CaraNet: context axial reverse attention network for segmentation of small medical objects,” in *Medical Imaging 2022: Image Processing*, I. Išgum and O. Colliot, Eds., SPIE, Apr. 2022, p. 11. doi: 10.1117/12.2611802.

[100] S. Poudel and S.-W. Lee, “Deep multi-scale attentional features for medical image segmentation,” *Appl. Soft Comput.*, vol. 109, p. 107445, Sep. 2021, doi: 10.1016/j.asoc.2021.107445.

[101] Y. Zhang, H. Liu, and Q. Hu, “TransFuse: Fusing Transformers and CNNs for Medical Image Segmentation,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12901 LNCS, pp. 14–24, 2021, doi: 10.1007/978-3-030-87193-2_2.

[102] 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,” 2020, pp. 451–462. doi: 10.1007/978-3-030-37734-2_37.

[103] 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,” *Int. J. Comput. Assist. Radiol. Surg.*, 2014, doi: 10.1007/s11548-013-0926-3.

[104] D. Vázquez, J. Bernal, F. Javier Sánchez, G. Fernández-Esparrach, A. M. López, A. Romero, M. Drozdzal, and A. Courville, “A Benchmark for Endoluminal Scene Segmentation of Colonoscopy Images,” *J. Healthc. Eng.*, vol. 2017, 2017, doi: 10.1155/2017/4037190.

[105] A. C. Society, “Colorectal Cancer Facts and Figures,” pp. 1–20, 2023. [Online]. Available: https://www.cancer.org/research/cancer-facts-statistics/colorectal-cancer-facts-figures.html

[106] S. Levenstein, Z. Li, S. Almer, A. Barbosa, P. Marquis, G. Moser, A. Sperber, B. Toner, and D. A. Drossman, “Predictors of inadequate bowel preparation for colonoscopy,” *Am. J. Gastroenterol.*, vol. 96, no. 6, pp. 1797–1802, Jun. 2001, doi: 10.1016/S0002-9270(01)02437-6.

[107] M. S. Cappell and D. Friedel, “The role of sigmoidoscopy and colonoscopy in the diagnosis and management of lower gastrointestinal disorders: Endoscopic findings, therapy, and complications,” *Medical Clinics of North America*, vol. 86, no. 6. Elsevier, pp. 1253–1288, Nov. 01, 2002. doi: 10.1016/S0025-7125(02)00077-9.

[108] E. J. Lai, A. H. Calderwood, G. Doros, O. K. Fix, and B. C. Jacobson, “The Boston bowel preparation scale: a valid and reliable instrument for colonoscopy-oriented research,” *Gastrointest. Endosc.*, vol. 69, no. 3, pp. 620–625, Mar. 2009, doi: 10.1016/j.gie.2008.05.057.

[109] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, “On the Properties of Neural Machine Translation: Encoder–Decoder Approaches,” 2015. doi: 10.3115/v1/w14-4012.

[110] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*,

2015.

[111] K. Pearson, “LIII. On lines and planes of closest fit to systems of points in space,” *London, Edinburgh, Dublin Philos. Mag. J. Sci.*, vol. 2, no. 11, pp. 559–572, Nov. 1901, doi: 10.1080/14786440109462720.

[112] G. Hinton and S. Roweis, “Stochastic neighbor embedding,” *Adv. Neural Inf. Process. Syst.*, 2003.

[113] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. D. Facebook, A. I. Research, Z. Lin, A. Desmaison, L. Antiga, O. Srl, and A. Lerer, “Automatic differentiation in PyTorch,” Oct. 2017.

[114] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[115] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, “Two-Stream Deep Feature Modelling for Automated Video Endoscopy Data Analysis,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, Oct. 2020, pp. 742–751. doi: 10.1007/978-3-030-59716-0_71.

[116] J. Zhou, L. Wu, X. Wan, L. Shen, J. Liu, J. Zhang, X. Jiang, Z. Wang, S. Yu, J. Kang, M. Li, S. Hu, X. Hu, D. Gong, D. Chen, L. Yao, Y. Zhu, and H. Yu, “A novel artificial intelligence system for the assessment of bowel preparation (with video),” *Gastrointest. Endosc.*, vol. 91, no. 2, pp. 428-435.e2, Feb. 2020, doi: 10.1016/j.gie.2019.11.026.

[117] 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,” Oct. 2020, [Online]. Available: http://arxiv.org/abs/2010.11929

[118] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, and A. Dosovitskiy, “MLP-Mixer: An all-MLP Architecture for Vision,” May 2021, Accessed: Jun. 22, 2021. [Online]. Available: http://arxiv.org/abs/2105.01601

[119] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016. doi: 10.1109/CVPR.2016.308.

[120] D. Tran, H. Wang, M. Feiszli, and L. Torresani, “Video Classification with Channel-Separated Convolutional Networks,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-October, pp. 5551–5560, Apr. 2019, doi: 10.1109/ICCV.2019.00565.

[121] J. Carreira and A. Zisserman, “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 4724–4733, May 2017, doi: 10.1109/CVPR.2017.502.

[122] C. Feichtenhofer, H. Fan, J. Malik, and K. He, “SlowFast Networks for Video Recognition,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-October, pp. 6201–6210, Dec. 2018, doi: 10.1109/ICCV.2019.00630.

[123] D. Rex, E. Rahmani, J. Haseman, G. Lemmel, S. Kaster, and J. Buckley, “Relative sensitivity of colonoscopy and barium enema for detection of colorectal cancer in clinical practice,” *Gastroenterology*, vol. 112, no. 1, pp. 17–23, Jan. 1997, doi: 10.1016/S0016-5085(97)70213-0.

[124] N. T. Duc, N. T. Oanh, N. T. Thuy, T. M. Triet, and V. S. Dinh, “ColonFormer: An Efficient Transformer Based Method for Colon Polyp Segmentation,” *IEEE Access*, vol. 10, pp. 80575–80586, 2022, doi: 10.1109/ACCESS.2022.3195241.

[125] X. Zhao, L. Zhang, and H. Lu, “Automatic Polyp Segmentation via Multi-scale Subtraction Network,” in *de Bruijne, M., et al. Medical Image Computing and Computer Assisted Intervention – MICCAI 2021. MICCAI 2021. Lecture Notes in Computer Science(), vol 12901. Springer*, 2021, pp. 120–130. doi: 10.1007/978-3-030-87193-2_12.

[126] D. Jha, P. H. Smedsrud, D. Johansen, T. De Lange, H. D. Johansen, P. Halvorsen, and M. A. Riegler, “A Comprehensive Study on Colorectal Polyp Segmentation with ResUNet++, Conditional Random Field and Test-Time Augmentation,” *IEEE J. Biomed. Heal. Informatics*, vol. 25, no. 6, pp. 2029–2040, 2021, doi: 10.1109/JBHI.2021.3049304.

[127] X. Zhao, H. Jia, Y. Pang, L. Lv, F. Tian, L. Zhang, W. Sun, and H. Lu, “M2SNet: Multi-scale in Multi-scale Subtraction Network for Medical Image Segmentation,” *arXiv*, pp. 1–10, Mar. 2023, [Online]. Available: http://arxiv.org/abs/2303.10894

[128] M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab, “An embedded recurrent neural network-based model for endoscopic semantic segmentation,” in *Proceedings of the 3rd International Workshop and Challenge on Computer Vision in Endoscopy (EndoCV 2021) co-located with the 18th IEEE International Symposium on Biomedical Imaging (ISBI 2021), Nice, France. (CEUR-WS.org, 2021)*, 2021, pp. 59–68.

[129] M. Haithami, A. Ahmed, I. Y. Liao, and H. Jalab, “Employing GRU to combine feature maps in DeeplabV3 for a better segmentation model,” *Nord. Mach. Intell.*, vol. 1, no. 1, pp. 29–31, 2021, doi: 10.5617/nmi.9131.

[130] H. Kvinge, S. Emerson, Tegan Jorgenson, Grayson Vasquez, T. Doster, and J. Lew, “In What Ways Are Deep Neural Networks Invariant and How Should We Measure This?,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2022, pp. 32816--32829.

[131] J. Zhang, H. Chao, A. Dhurandhar, P.-Y. Chen, A. Tajer, Y. Xu, and P. Yan, “When Neural Networks Fail to Generalize? A Model Sensitivity Perspective,” *Proc. AAAI Conf. Artif. Intell.*, vol. 37, no. 9, pp. 11219–11227, Jun. 2023, doi: 10.1609/aaai.v37i9.26328.

[132] J. Gauthier, “Conditional generative adversarial nets for convolutional

face generation," *Cl. Proj. Stanford CS231N convolutional neural networks Vis. recognition, Winter semester*, 2014.

[133] "Colon Polyp and Cancer," Digestive & Liver Health Specialist. Accessed: Oct. 20, 2023. [Online]. Available: https://thegidocs.com/patient-handouts/colon-polyp-and-cancer/

[134] V. Thambawita, P. Salehi, S. A. Sheshkal, S. A. Hicks, H. L. Hammer, S. Parasa, T. de Lange, P. Halvorsen, and M. A. Riegler, "SinGAN-Seg: Synthetic training data generation for medical image segmentation," *PLoS One*, vol. 17, no. 5, p. e0267976, May 2022, doi: 10.1371/journal.pone.0267976.

[135] T. R. Shaham, T. Dekel, and T. Michaeli, "Singan: Learning a generative model from a single natural image," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4570–4580.

[136] L. Gatys, A. Ecker, and M. Bethge, "A Neural Algorithm of Artistic Style," *J. Vis.*, vol. 16, no. 12, p. 326, Sep. 2016, doi: 10.1167/16.12.326.

[137] L. Perez and J. Wang, "The Effectiveness of Data Augmentation in Image Classification using Deep Learning," Dec. 2017, Accessed: Jan. 07, 2020. [Online]. Available: http://arxiv.org/abs/1712.04621

[138] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1–4, pp. 259–268, Nov. 1992, doi: 10.1016/0167-2789(92)90242-F.

[139] C. M. Bishop, *Pattern recognition and machine learning*. New York: springer, 2006.

[140] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," pp. 1–15, Dec. 2014, [Online]. Available: http://arxiv.org/abs/1412.6980

[141] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, Oct. 2019, pp. 1314–1324. doi: 10.1109/ICCV.2019.00140.

[142] 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," 2014, pp. 740–755. doi: 10.1007/978-3-319-10602-1_48.

[143] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, "A review of semantic segmentation using deep neural networks," *Int. J. Multimed. Inf. Retr.*, vol. 7, no. 2, pp. 87–93, Jun. 2018, doi: 10.1007/s13735-017-0141-z.

[144] J. Bernal, J. Sanchez, and F. Vilarino, "Impact of image preprocessing methods on polyp localization in colonoscopy frames," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, Jul. 2013, pp. 7350–7354. doi: 10.1109/EMBC.2013.6611256.

[145] J. Bernal, J. M. Núñez, F. J. Sánchez, and F. Vilariño, "Polyp Segmentation Method in Colonoscopy Videos by Means of MSA-DOVA

Energy Maps Calculation," 2014, pp. 41–49. doi: 10.1007/978-3-319-13909-8_6.

[146] J. Bernal, D. Gil, C. Sánchez, and F. J. Sánchez, "Discarding Non Informative Regions for Efficient Colonoscopy Image Analysis," 2014, pp. 1–10. doi: 10.1007/978-3-319-13410-9_1.

[147] B. Li and M. Q.-H. Meng, "Texture analysis for ulcer detection in capsule endoscopy images," *Image Vis. Comput.*, vol. 27, no. 9, pp. 1336–1342, Aug. 2009, doi: 10.1016/J.IMAVIS.2008.12.003.

[148] A. R. Hassan and M. A. Haque, "Computer-aided gastrointestinal hemorrhage detection in wireless capsule endoscopy videos," *Comput. Methods Programs Biomed.*, vol. 122, no. 3, pp. 341–353, Dec. 2015, doi: 10.1016/J.CMPB.2015.09.005.

[149] S. Seguí, M. Drozdzal, G. Pascual, P. Radeva, C. Malagelada, F. Azpiroz, and J. Vitrià, "Generic feature learning for wireless capsule endoscopy analysis," *Comput. Biol. Med.*, vol. 79, pp. 163–172, Dec. 2016, doi: 10.1016/J.COMPBIOMED.2016.10.011.

[150] S. V. Georgakopoulos, D. K. Iakovidis, M. Vasilakakis, V. P. Plagianakos, and A. Koulaouzidis, "Weakly-supervised Convolutional learning for detection of inflammatory gastrointestinal lesions," in *2016 IEEE International Conference on Imaging Systems and Techniques (IST)*, IEEE, Oct. 2016, pp. 510–514. doi: 10.1109/IST.2016.7738279.

[151] E. Rodriguez-Diaz and S. K. Singh, "Sa2029 Computer-Assisted Interpretation of the NICE Criteria for Colorectal Polyps using Near-Focus Narrow-Band Imaging," *Gastroenterology*, vol. 150, no. 4, p. S434, Apr. 2016, doi: 10.1016/S0016-5085(16)31506-2.

[152] A. Kage, M. Raithel, S. Zopf, T. Wittenberg, and C. Münzenmayer, "Narrow-band imaging for the computer assisted diagnosis in patients with Barrett's esophagus," in *Medical Imaging 2009: Computer-Aided Diagnosis*, N. Karssemeijer and M. L. Giger, Eds., International Society for Optics and Photonics, Feb. 2009, p. 72603S. doi: 10.1117/12.812257.

[153] D. Boschetto, G. Gambaretto, and E. Grisan, "Automatic classification of endoscopic images for premalignant conditions of the esophagus," in *Medical Imaging 2016: Biomedical Applications in Molecular, Structural, and Functional Imaging*, B. Gimi and A. Krol, Eds., International Society for Optics and Photonics, Mar. 2016, p. 978808. doi: 10.1117/12.2216826.

[154] D.-X. Xue, R. Zhang, Y.-Y. Zhao, J.-M. Xu, and Y.-L. Wang, "Fully convolutional networks with double-label for esophageal cancer image segmentation by self-transfer learning," in *Ninth International Conference on Digital Image Processing (ICDIP 2017)*, C. M. Falco and X. Jiang, Eds., Jul. 2017, p. 104202D. doi: 10.1117/12.2282000.

[155] E. Grisan, E. Veronese, G. Diamantis, C. Trovato, C. Crosta, and G. Battaglia, "Computer Aided Diagnosis of Barrett's Esophagus Using Confocal Laser Endomicroscopy: Preliminary Data," *Dig. Liver Dis.*,

vol. 44, no. 4, pp. S147–S148, Mar. 2012, doi: 10.1016/S1590-8658(12)60411-3.

[156] E. Veronese, E. Grisan, G. Diamantis, G. Battaglia, C. Crosta, and C. Trovato, "Hybrid patch-based and image-wide classification of confocal laser endomicroscopy images in Barrett's esophagus surveillance," *Proc. - Int. Symp. Biomed. Imaging*, no. c, pp. 362–365, 2013, doi: 10.1109/ISBI.2013.6556487.

[157] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002, doi: 10.1109/TPAMI.2002.1017623.

[158] N. Ghatwary, A. Ahmed, X. Ye, and H. Jalab, "Automatic grade classification of Barretts Esophagus through feature enhancement," in *Medical Imaging 2017: Computer-Aided Diagnosis*, S. G. Armato and N. A. Petrick, Eds., Mar. 2017, p. 1013433. doi: 10.1117/12.2250364.

[159] J. Hong, B. Park, and H. Park, "Convolutional neural network classifier for distinguishing Barrett's esophagus and neoplasia endomicroscopy images," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, Jul. 2017, pp. 2892–2895. doi: 10.1109/EMBC.2017.8037461.

[160] N. Ghatwary, A. Ahmed, E. Grisan, H. Jalab, L. Bidaut, and X. Ye, "In-vivo Barrett's esophagus digital pathology stage classification through feature enhancement of confocal laser endomicroscopy," *J. Med. Imaging*, vol. 6, no. 01, p. 1, Mar. 2019, doi: 10.1117/1.JMI.6.1.014502.

[161] F. van der Sommen, S. Zinger, E. J. Schoon, and P. H. N. de With, "Computer-aided detection of early cancer in the esophagus using HD endoscopy images," in *Medical Imaging 2013: Computer-Aided Diagnosis*, C. L. Novak and S. Aylward, Eds., Feb. 2013, p. 86700V. doi: 10.1117/12.2001068.

[162] F. van der Sommen, S. Zinger, E. J. Schoon, and P. H. N. de With, "Supportive automatic annotation of early esophageal cancer using local gabor and color features," *Neurocomputing*, vol. 144, pp. 92–106, Nov. 2014, doi: 10.1016/j.neucom.2014.02.066.

[163] F. van der Sommen, S. Zinger, E. J. Schoon, and P. H. N. de With, "Sweet-spot training for early esophageal cancer detection," G. D. Tourassi and S. G. Armato, Eds., International Society for Optics and Photonics, Mar. 2016, p. 97851B. doi: 10.1117/12.2208114.

[164] M. H. A. Janse, F. van der Sommen, S. Zinger, E. J. Schoon, and P. H. N. de With, "Early esophageal cancer detection using RF classifiers," in *Medical Imaging 2016: Computer-Aided Diagnosis*, G. D. Tourassi and S. G. Armato, Eds., International Society for Optics and Photonics, Mar. 2016, p. 97851D. doi: 10.1117/12.2208583.

[165] H. Matsunaga, H. Omura, R. Ohura, and T. Minamoto, "Daubechies wavelet-based method for early esophageal cancer detection from

flexible spectral imaging color enhancement image," in *Advances in Intelligent Systems and Computing*, vol. 448, Springer, Cham, 2016, pp. 939–948. doi: 10.1007/978-3-319-32467-8_81.

[166] L. Souza, C. Hook, J. P. Papa, and C. Palm, "Barrett's Esophagus Analysis Using SURF Features," in *Informatik aktuell*, 2017, pp. 141–146. doi: 10.1007/978-3-662-54345-0_34.

[167] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer, Berlin, Heidelberg, 2006, pp. 404–417. doi: 10.1007/11744023_32.

[168] R. Mendel, A. Ebigbo, A. Probst, H. Messmann, and C. Palm, "Barrett's esophagus analysis using convolutional neural networks," in *Informatik aktuell*, Springer Vieweg, Berlin, Heidelberg, 2017, pp. 80–85. doi: 10.1007/978-3-662-54345-0_23.

[169] L. A. De Souza, L. C. S. Afonso, C. Palm, and J. P. Papa, "Barrett's Esophagus Identification Using Optimum-Path Forest," in *Proceedings - 30th Conference on Graphics, Patterns and Images, SIBGRAPI 2017*, IEEE, Oct. 2017, pp. 308–314. doi: 10.1109/SIBGRAPI.2017.47.

[170] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004, doi: 10.1023/B:VISI.0000029664.99615.94.

[171] L. Serpa-Andrade, V. Robles-Bykbaev, L. González-Delgado, and J. L. Moreno, "An approach based on Fourier descriptors and decision trees to perform presumptive diagnosis of esophagitis for educational purposes," in *2015 IEEE International Autumn Meeting on Power, Electronics and Computing, ROPEC 2015*, IEEE, Nov. 2016, pp. 1–5. doi: 10.1109/ROPEC.2015.7395123.

[172] S. Van Riel, F. Van Der Sommen, S. Zinger, E. J. Schoon, and P. H. N. de With, "Automatic Detection of Early Esophageal Cancer with CNNS Using Transfer Learning," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, IEEE, Oct. 2018, pp. 1383–1387. doi: 10.1109/ICIP.2018.8451771.

[173] N. Ghatwary, M. Zolgharni, and X. Ye, "Early esophageal adenocarcinoma detection using deep learning methods," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 14, no. 4, pp. 611–621, Apr. 2019, doi: 10.1007/s11548-019-01914-4.

[174] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014. doi: 10.1109/CVPR.2014.81.

[175] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015. doi: 10.1109/ICCV.2015.169.

[176] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A.

C. Berg, “SSD: Single shot multibox detector,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[177] Y. Horie, T. Yoshio, K. Aoyama, S. Yoshimizu, Y. Horiuchi, A. Ishiyama, T. Hirasawa, T. Tsuchida, T. Ozawa, S. Ishihara, Y. Kumagai, M. Fujishiro, I. Maetani, J. Fujisaki, and T. Tada, “Diagnostic outcomes of esophageal cancer by artificial intelligence using convolutional neural networks,” *Gastrointest. Endosc.*, vol. 89, no. 1, pp. 25–32, Jan. 2019, doi: 10.1016/j.gie.2018.07.037.

[178] A. Ebigbo, R. Mendel, A. Probst, J. Manzeneder, F. Prinz, L. A. de Souza Jr., J. Papa, C. Palm, and H. Messmann, “Real-time use of artificial intelligence in the evaluation of cancer in Barrett’s oesophagus,” *Gut*, p. gutjnl-2019-319460, Sep. 2019, doi: 10.1136/gutjnl-2019-319460.

[179] N. Ghatwary, X. Ye, and M. Zolgharni, “Esophageal Abnormality Detection Using DenseNet Based Faster R-CNN With Gabor Features,” *IEEE Access*, vol. 7, pp. 84374–84385, 2019, doi: 10.1109/ACCESS.2019.2925585.

[180] D. Y. Liu, H. X. Jiang, N. N. Rao, C. S. Luo, W. J. Du, Z. W. Li, and T. Gan, “Computer aided annotation of early esophageal cancer in gastroscopic images based on Deeplabv3+ Network,” in *ACM International Conference Proceeding Series*, New York, New York, USA: ACM Press, Aug. 2019, pp. 56–61. doi: 10.1145/3354031.3354046.

[181] Z. Zhang, L. Bai, P. Ren, and E. R. Hancock, “High-order graph matching kernel for early carcinoma EUS image classification,” *Multimed. Tools Appl.*, vol. 75, no. 7, pp. 3993–4012, Apr. 2016, doi: 10.1007/s11042-015-3108-1.

[182] S. Klomp, F. van der Sommen, A.-F. Swager, S. Zinger, E. J. Schoon, W. L. Curvers, J. J. Bergman, and P. H. N. de With, “Evaluation of image features and classification methods for Barrett’s cancer detection using VLE imaging,” in *Medical Imaging 2017: Computer-Aided Diagnosis*, S. G. Armato and N. A. Petrick, Eds., International Society for Optics and Photonics, Mar. 2017, p. 101340D. doi: 10.1117/12.2253860.

[183] A. F. Swager, F. van der Sommen, S. R. Klomp, S. Zinger, S. L. Meijer, E. J. Schoon, J. J. G. H. M. Bergman, P. H. de With, and W. L. Curvers, “Computer-aided detection of early Barrett’s neoplasia using volumetric laser endomicroscopy,” *Gastrointest. Endosc.*, vol. 86, no. 5, pp. 839–846, Nov. 2017, doi: 10.1016/j.gie.2017.03.011.

[184] D. K. Chan, L. Zakko, K. H. Visrodia, C. L. Leggett, L. S. Lutzke, M. A. Clemens, J. D. Allen, M. A. Anderson, and K. K. Wang, “Breath Testing for Barrett’s Esophagus Using Exhaled Volatile Organic Compound Profiling With an Electronic Nose Device,” *Gastroenterology*, vol. 152, no. 1, pp. 24–26, Jan. 2017, doi: 10.1053/j.gastro.2016.11.001.

[185] C. Li, C. Shi, H. Zhang, Y. Chen, and S. Zhang, “Multiple instance

learning for computer aided detection and diagnosis of gastric cancer with dual-energy CT imaging," *J. Biomed. Inform.*, vol. 57, pp. 358–368, Oct. 2015, doi: 10.1016/J.JBI.2015.08.017.

[186] T. J. Muldoon, N. Thekkek, D. Roblyer, D. Maru, N. Harpaz, J. Potack, S. Anandasabapathy, and R. Richards-Kortum, "Evaluation of quantitative image analysis criteria for the high-resolution microendoscopic detection of neoplasia in Barrett's esophagus," *J. Biomed. Opt.*, vol. 15, no. 2, p. 026027, 2010, doi: 10.1117/1.3406386.

[187] D. Marc Peter, F. A. Aldo, and O. Cheng Soon, *Mathematics for machine learning*. Cambridge University Press, 2020.

Appendices

Appendix A

Selected papers related to digestive system are summarized in Table 28. It contains deep learning methods as well as machine learning methods. Furthermore, it includes endoscopic and non-endoscopic domains.

Table 28 Papers related to the application of AI in the digestive system.

Capsule Endoscopy						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[147]	- Presented a new texture-based protocol for ulcer regions - curvelets - Local Binary Patterns (LBP)	- None	- Multilayer Perceptron Network (MLP) - SVM	- MLP in YCbCr "Best combination": 92.37% accuracy 91.46% specificity 93.28% sensitivity	100 **private** images. Each image is segmented into patches and each patch is classified either ulcer or not.	4- fold cross validation
[148]	- proposed real time bleeding detection - developed texture feature descriptor that operates on the Normalised Grey Level Co-occurrence Matrix (NGLCM) in the magnitude spectrum of the images.	- None	-SVM	99.19% Accuracy 99.41% Sensitivity 98.95% Specificity	- 600 bleeding and 600 non-bleeding images used for training. - 860 bleeding and 860 non-bleeding images used for testing.	Cross validation
[149]	- Identify small intestine motility type using Deep CNN - the motility types are; wall, wrinkles, bubbles, turbid, clear blob and undefined. - It showed a better performance than the handcrafted features proposed by others.	- None	- simple CNN that has 3 conv, 3 pool and 3 FC	- Mean Accuracy 96%	- 100,000 annotated images. - 10,000 testing images.	Cross Validation

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[150]	- Detect inflammatory lesions in the Gastrointestinal tract. - They investigate the visibility of training a CNN using image-level annotations (e.g. contain inflammation or not) rather than graphical annotation (i.e. pixel-based). - Proposed a weakly-supervised learning technique based on CNN that uses only image-level semantic annotations for the training process.	-None	- Weakly supervised learning CNN - 5 conv layers and 4 max-pooling followed by 2 FC	- 90% Accuracy - 92% Sensitivity - 88% Specificity.	- KID "public dataset" - 227 graphically annotated images of inflammatory lesions and 599 normal images of the GI tract. - Training (200 normal & 200 abnormal) - Testing (27 normal & 27 abnormal)	Cross Validation
Narrow Band Imaging (NBI) & Near-Focus (NF-NBI)						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[151]	- They used colour space extracted from NF-NBI images - They applied crypt-space colour segmentation And Discrete Wavelet Transform	- None	- SVM	86% sensitivity and specificity	56 images (16 non-neoplastic polyps)	LOOCV
[152]	- Collection of Features were extracted from NBI images: - Co-occurrence matrices, summation and difference of histogram, statistical geometric and Gabor Filters.	- None	- Euclidian Distance	Accuracy [85% - 92%] for a combination of features. Epithelium 97% (202 images), Cardiac 91% (78 images), and BE 74% (46 images)	326 regions of Interest annotated and classified between: 1- Epithelium 2- Cardiac mucosa 3- BE	LOOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[153]	- The employed super-pixels segmentation technique. - They extracted 8 features from each super-pixels. -3 features are calculated as mean intensities of each colour channel -3 features stand for mean intensities of the red-channel with the application of three different morphological filters (top-hat, entropy and range filters) - 2 features are related to the contrast and homogeneity of the super-pixels.	- None	- Random Forest	83.9% Accuracy 79.2% Sensitivity 87.3% Specificity	116 NBI images private from Oncologico Veneto. Detect the metaplasia region within the images.	10-fold-cross validation
[154]	- The aim is to evaluate the feasibility of automated classification of intrapapillary capillary loops (IPCLs) to improve the detection of esophageal squamous cell carcinoma (ESCC). - A double-labelling fully convolutional network (FCN) was developed for image segmentation.	- Augmentation methods were used but they didn't mention which type of augmentation.	FCN "CNN"	- mean diagnostic accuracy 89% at the lesion level and 93% at the pixel level. Performed significantly better than endoscopists.	-Private - 1383 NBI-Magnified images were studied (207 type A, 970 type B1, and 206 type B2)	
Confocal Laser Endomicroscopy (CLE)						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[155]	- Contrast value and a multiscale texture analysis was obtained with rotation invariant local binary pattern (RI-LBP) from each sub-image. - Classify to either **GM** or **IM**	- None	- SVM	- 98.85% Sensitivity - 65.22% Specificity in the detection of IM. - By ROC curve, trade-off can be made (96.5% Sensitivity, 95.6% Specificity)	285 images (262 images showing IM and 23 GM). Using a simple voting scheme, the image is classified either GM or IM.	LOOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[156]	- Block Features where extracted to Identify the image either IM or Other. - Image Features to identify GM or NPL - Block Features: multiscale local binary pattern (LBP), (moments, percentiles, entropy). - Image Features: fractal dimension of the bright pixels, computed with the box counting method [157]	- None	-SVM	Sensitivity: - 96% Gastric Metaplasia (GM) - 95% Intestinal Metaplasia (IM) - 100% Neoplasia (NP) "cancerous"	Total 337 - GM 23 images - IM 263 images - NPL 51 images	LOOCV
[158]	- Three stages classification model was developed to distinguish between **IM**, **GM** and **NPL** - The image is first enhanced by applying fractional differential and fractional integration in the wavelet sub-bands. - Gray Level Co-occurrence Matrices (GLCM), Fractal Texture Features, Fuzzy Local Binary Patter (FLBP), Intensity Features and Wavelet Features. - For each classification stage, subset of these features was employed	- None	-SVM	- Accuracy 90.45% - 98.8% for discriminating NPL from others -96.7% to separate IM from GM	- 32 patients - 262 images Public images. - GM 172 images - IM 30 images - NPL 60 images	LOOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[159]	- They investigate the visibility of using CNN to classify endomicroscopy imaging data of BE. - propose 5 simple distortions (random scaling, random flip, random rotation, random brightness, and random contrast - Designed the CNN architecture - They acknowledge the lack of data, so he employed simple augmentation techniques. However, they didn't report the effectiveness of the used augmentation methods.	- They used random gaussian scaling with a mean of 1.0 and standard deviation of 0.2. - The scaled image is then randomly flipped vertically or horizontally and then they were rotated in counter-clockwise direction by 0º, 90º, 180º and 270º - They added random number under 30 for each pixel to adjust the brightness - the image contrasts were adjusted with the contrast factor between 0.8 and 1.2.	-CNN composed of 4 conv layers, 2 max-pooling layers 2 FC layers	- Total Accuracy 80.77% (26 images are used for testing)	- (ISBI) 2016 challenge database (public) - 262 images used (236 for training and 26 images for testing) 26 images -17 IM images -4 GM images -5 NPL images	Cross Validation
[160]	- Classify images into Normal Squamous NS, gastric metaplasia (GM), intestinal metaplasia (IM), and neoplasia (NPL) - Enhance the image in the DWT domain by fractional differentiation (FD) and fractional integration (FI), then features were extracted - Multiscale Pyramid with Rotation Invariant Local Binary Pattern (MP-RLBP), Maximally Stable Extremal Regions (MSER), Gray Level Co-occurrence Matrix (GLCM) features (entropy, contrast, and homogeneity), fractal features (i.e. dimension using box-counting, mean gray level, and pixel count) , and Fuzzy Local Binary Pattern (FLBP)	- None	-SVM -Random Forest	- SVM achieved 96% Accuracy - Random Forest 91% Accuracy	- 557 images - (IM 402 images, GM 41 images, NPL 68 images, NS 45 images)	LOPOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

High Definition White Light Endoscopy (HD-WLE)						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[161]	1- pre-processing the image (Colour Transformation + (DWT)) 2- feature extraction (colour histogram + Gabor features). Dimensionality reduction 3- classification of a patch to either tumorous or normal. (50*50 pixel) - Detect if the image has early cancer and where it resides. -Detect and locate early cancer in Oesophagus to support physicians in finding early stage cancer.	-None	-SVM. Different parameters and kernels were used	95% Accuracy with 99% Area under the curve (AUC)	- 66 patients - ?? images - Each image is segmented into patch and each patch is classified either tumorous or normal	10-fold cross-validation
[162]	- Extract local colour/texture features based on the original and Gabor-filtered image. -Detect and locate early cancer in Barrett's Oesophagus to support physicians in finding early stage cancer.	-None	-SVM	-95% Recall -75% Precision	-Private -64 images -7 patients EAC/15 without EAC	LOPOC
[163]	- Based on the previous research published in 2014. - They argued that the gastrointestinal tract cancerous tissue should not be treated as a binary problem - Sweet spot metric is proposed for the training phase "SST" and the Jaccard Golden Standard "JIGS" a metric able to handle multiple annotations. - They showed that these two metrics increased the performance of a detection algorithm of early neoplastic lesions in BE by 10% using F1-score	-None	-SVM	They showed that the system performance of a detection algorithm can be increased by up to 10% of F1	- MICCAI dataset - 100 endoscopic images from 39 patients	---

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[164]	- Enhance a CAD system based on their previous work [162] by introducing Random Forest, thereby introducing a measure of confidence for the detected regions. - (ROI) is determined by removing borders, reflections, and the lumen. - features extraction from ROI then classification and finally annotation.	-None	-Random Forest	- 75% Accuracy - 90% Recall	-100 images of 39 patients. 50 images showing cancerous. - Annotated by 5 gastro experts	LOPOC
[165]	- proposed a method to diagnose early Esophageal Cancer from images - they employed the Dyadic Wavelet Transform (DYWT) and fractal dimension	HDE LBI FICE	- Unsupervised	No quantification data	23 patients with EC	- no Data found
[166]	- Detect Adenocarcinoma in BE patients - Used Group of features called Speeded Up Robust Features (SURF) [167]	-None	-SVM	- Image-Based accuracy: 77% Sensitivity 82% specificity - Region-Based accuracy: 89.6% Sensitivity 95.1% Specificity	- MICCAI - 100 public image images from EndoVis Challenge	LOPOCV
[168]	- Early detection of adenocarcinoma in the oesophagus of BE patients - A deep convolutional neural network is adapted to the data using a transfer learning approach. - The image is divided into patches and a deep learning is trained to produce a probability for a cancer. - The image is assigned as cancerous if a one patch is detected as cancerous with a predefined threshold.	- Since the cancerous patches is smaller than non-cancerous, data augmentation was employed to solve the imbalance data. 1-Randomly, 30% of the labelled patch are selected and rotated by:90◦, 180◦ or 270◦.	- Deep CNN -ResNet with transfer learning. - The ResNet was initialised with the parameters learned on the ImageNet dataset.	Sensitivity 94% Specificity 88%	100 public image images from EndoVis Challenge (MICCAI) - 50 images BE and 50 images adenocarcinoma (Cancerous).	LOPOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[169]	- The aim is to detect BE lesions - They used SURF (Speeded Up Robust Features[167]) AND SIFT (Scale-Invariant Feature Transform [170]) *Computer Vision Techniques - For comparisons, they used Optimum Path Forest classifier and SVM	- None	- SVM - OPF	- OPF outperform SVM. - OPF results: SURF: 73% Sensitivity 78% Specificity 73% Accuracy. SIFT: 73% Sensitivity 80% Specificity 73% Accuracy.	- 100 public image images from EndoVis Challenge (MICCAI). Some of them are normal and some has BE lesion.	Cross-Validation
[171]	- Detect the esophagitis by analysing esophageal irregularities (the Z-line). - The true image is converted to gray. - To segment the z-line, the system and a user define the segmentation parameters using the watershed algorithm. -The statistical Hu Momentum is collected from the extracted region and Fourier transform is applied on shape signature of the Z-line.	-None	- K-NN - Random Forest	(Fourier + RF) Best Combinations: 86% Sensitivity 72% Specificity 80% Accuracy	10 healthy tissue images and 16 ill tissue images.	Cross-Validation
[172]	- The goal is to achieve real-time performance in order to work towards a clinical application - Detects Esophageal cancer EC using Transfer Learning of CNN - Intermediate layers of the network are used as features in traditional classifiers. - Sliding window are used to locate the cancer - the detection and annotation at 2 frames per second (fbs), which is suitable tame for real-time application.	- None	- Transfer Learning - CNN	-Using VGG16 and SVM: ~92% AUC using AlexNet with SVM at window size of 200*150 pixels.	- MICCAI dataset - 39 patients - 100 images	LOPOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[173]	- The aim is to study the feasibility of the state-of-art CNN object detection algorithms to detect Esophageal abnormalities. - The tested CNNs are; 1) Regional-based CNN "R-CNN" [174] 2) Fast R-CNN [175] 3) Faster R-CNN [9] 4) single shot multibox detector "SSD" [176]	-flipping along the axial plane and rotation in different angles with 90◦, 180◦ and 270◦. - The effect of these augmentation is not reported nor the number of generated images.	- R-CNN - Fast R-CNN - Faster R-CNN - SSD	- The best results are achieved by SSD: - cross-validation: 93% specificity 93% sensitivity - 5-fold-cv: 88% specificity 90% sensitivity - LOPOCV: 92% specificity 96% sensitivity	- MICCAI dataset -100 images annotated by 5 experts.	- 60%/40% cross-validation. - 5-fold-cv - LOPOCV
[49] . This work is an extension to his previous work [168]	- Two tasks: 1) Classification 2) Segmentation "delineation" - ResNet consist of 100 layers. - Small patches were used in the training stage. The patches were augmented "i.e. rotation mirroring …etc" - high/low grade dysplasia are not included in the study.	- None	- ResNet	- Augsburg data WLE: 97% Sensitivities 88% Specificities - Augsburg data NBI: 94% Sensitivities 80% Specificities - MICCAI data WLE: 92% Sensitivities 100% Specificities - Using a significant test "McNemar", the system outperformed 11 endoscopists in Sensitivity, Specificity or both "Augsburg dataset".	2 databases 1) Augsburg "their data". 148 HD images WLE+NBI (33 early EAC and 41 non-neoplastic Barrett's mucosa) 2) MICCAI "public"	- LOPOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[177] "15 MD authors"	- The aim is to test the feasibility of Deep learning to detect Squamous Cell Carcinoma and Adenocarcinoma. - They employed SSD using Caffee Deep learning Framework	- None	-Single Shot Multi-Box "SSD" using Caffee Framework	- 98% Sensitivity in detection of cancer (patient based). - 98% Accuracy in distinguish between superficial from advance cancer (from all cancer images detected by the system correctly). ---------- However, for the image-based classification for both WLE+NBI: -79.1 Accuracy - 77% Sensitivity - 79% Specificity	- Private dataset - 384 patients. - 8428 training images - 1118 testing images (558 WLE & 560 NBI). 47 patients with 49 esophageal cancer (41 SCC and 8 adenocarcinomas) and 50 non-esophageal cancers patients.	Cross-Validation
[6] Real Time*	-hand crafted features of colour and texture. (I Need to read it)	- None		Identify the images to neoplastic or non-neoplastic - 95% sensitivity - 85% specificity	40 images of Barrett's cancer and 20 images non-dysplastic BE.	
[178] Real Time* Clinical version based on the above paper [49] and [168].	- Based on CNN and ResNet "101 layers" architecture, an encoder-decoder network was adapted. DeepLab V.3+ was used. - The system takes random images from the real-time video and provide a prediction of the probability of cancer.	- None	-CNN - ResNet	- 100% Sensitivity - 83.7% Specificity "outperformed human endoscopists"	- Trained on 129 images from their database "Augsburg" - Tested on 62 images (36 of early EAC and 26 of normal BE) - Classification: cancer vs non-cancer and annotations.	- --

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[179]	- They proposed deep learning method based on Faster R-CNN to automatically detect abnormalitites in the oesophagus from WLE images. - The system is based on a combintion of Gabor handcrafted features with the CNN features. - DenseNets architecture is emraced to extrract the CNN features.	- Simple augmentation techniques were applied to increase the dataset such as random rotation, flipping, stretching horizantaly and vertically.	- Dense Nets	- Kvasir: 90.2% Recall 92.1% Precision - MICCA: 95% Recall 91% Precision	- MICCA 2015 dataset (100 images) that contains EAC lesions - Kvasir dataset (1000 images) that contains Esophigities “precancerouse stage”	- Kvasir : cross-Validation(50% training ,10% validation and 40% testing) - MICCAI: LOPOCV (10% are used for validation and the rest for training)
[180]	- They employed Deeplabv3+ network for preliminary prediction of early esophageal cancer, then they used morphology with different radiuses to finalise the annotations.	- Simple Augmentation were adopted to increase the stabiltiy: 1- Rotation by 90,180 or 270 degree 2- Flipping 3- Increase/decrease brightness by 25%	- Deeplabv3+	- Precesion: 78% - Recall: 77.5% - DSC = 74%.	- Private database (3190 images of 732 patients) -500 images were used as testing (15% of the whole dataset)	- Cross-Validation

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
Flexible spectral Imagining Colour Enhancement (FICE)						
[165]	- Detect Early cancer EC by first converting the image into L*a*b*. - Only *a is processed. - The Daubechies Wavelet Transform (DWT) is calculated in non-overlapped blocks "64*64 pixels" - Detection is done by applying a threshold to the histogram.	-None	-Hard Coded	- None	The results are illustrated in 5 images. No quantification of the results	
Endoscopic ultrasonography (EUS)						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[181]	- Graph based algorithm - The texture features are extracted from the spectral domain of the images. " Scale-invariant Feature Transform - SIFT" - The texture features are represented as a graph where the nodes represent pixels' feature and the edge represent the similarity between the gray-level/local-features of the images. - The similarity was provided by a high-order graph matching of the texture features.	- None	- SVM "new kernel based on graphics matching for EUS images is designed"	- 93% Over all accuracy - For EC: Accuracy 89% Sensitivity 94% Specificity 95%	1210 EUS images. (66 with early cancer and 91 without). The images are classified into (early esophageal carcinoma, normal and leiomyoma tissues)	10-fold-cross validation
Volumetric Laser Endoscopy (VLE)						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[182]	- Proposed three new features based on the classic Haralick features. - Benchmarking of machine learning and feature extraction techniques.	- None	-SVM	- Receiver Operating Characteristic (ROC) 0.95 - Mod. GLCM achieved: - 92% Accuracy - 90% Sensitivity - 93% Specificity -95% AUC	60 images. 30 dysplastic BE images and 30 non-dysplastic BE images.	- 10-fold-cross validation.

Table 28 Papers related to the application of AI in the digestive system, continued.

Ref.	Methodology	Augmentation	Classifier	Results	Database	Validation Protocol
[183]	- Conducted VLE features comparisons. - Generic image analysis features (Gray-level co-occurrence matrix, Local Binary patterns, Histogram of oriented gradients, and wavelet transform) - Clinically inspired features "Proposed" (Layering, Signal Intensity Distribution, and signal decay statistics)	- None	- SVM -Discriminant Analysis - AdaBoost - Random Forest - K-Nearest Neighbours - Naïve Bayes - Linear Regression - Logistic Regression	- layering and signal decay statistic show an optimal performance Using AdaBoost. An area under the receiver operating characteristic curve (AUC) of 91%. 90% Sensitivity 93% Specificity.	60 images. - 30 nondysplastic BE and 30 high-grade dysplasia/early adenocarcinoma images	- LOOCV
E- Nose						
Ref	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[184]	- The goal is to develop a technology that discriminates the subtle differences of volatile organic compounds (VOCs), which differentiate the smell of diseases. - Identify patients who has BE by analysing the breath using e-nose - VOC profiles were introduced into an artificial neural network.	- None	- NN	- 81% Accuracy - 80% Specificity - 82% Sensitivity - Area Under Curve (AUC) 79%	- 122 patient. 66 Volatile Organic Compounds (VOC) of patients presenting BE and 56 VOC's pf patients without BE	Leave-some-out cross validation (LSOCV)
CT scan						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[185]	- They used Multiple Instance Learning method (MIL) for the identification of tumor invasion depth of gastric cancer with dual-energy CT imaging. - For instance-level features, a proposed Citation-KNN method is used to solve the ambiguity in assigning labels to selected patches.	- None	- Multiple Instance Learning (MIL)	76.9% total Accuracy	----	- LOOCV

Table 28 Papers related to the application of AI in the digestive system, continued.

High-Resolution Micro-Endoscope						
Ref.	**Methodology**	**Augmentation**	**Classifier**	**Results**	**Database**	**Validation Protocol**
[186]	- combination of colour & texture features were proposed - Principal Component Analysis were employed to reduce the features space	- None	-SVM	96.48% accuracy.	129 sites composed of 16 HMRE images.	Cross-validation

Appendix B

Chapter 3 provided the average across 2-fold cross-validation of 3 differently shuffled dataset. In order to facilitate analysis and comparison with future research, the average scores of 2-fold cross-validation for each individual dataset are provided in Table 29, Table 30, and Table 31. These tables correspond to the datasets listed in Table 2 Table 3, and Table 4, respectively.

Table 29 Validation results Dataset1. Dataset1 consists of 2-fold cross-validation. The best results highlighted.

Models (Avg. Dataset1)	**Precision**	**Recall**	**F1-score**	**Accuracy**
InceptionV3 [119]	70.85	71.28	69.64	71.28
VGG11 [110]	72.37	71.66	70.1	71.66
DenseNet [54]	70.4	66.86	64.91	66.86
ViT [117]	71.48	69.25	68.58	69.25
ResNet50 [7]	69.75	67.42	65.07	67.42
MLP_Mixer [118]	71.79	67.58	67.62	67.58
InceptionV3 +TL	74.95	73.77	71.27	73.77
VGG11 +TL	78.09	79.15	76.74	79.15
DenseNet +TL	77.45	77.76	76.17	77.76
ViT +TL	83.08	80.89	80.41	80.89
ResNet50 +TL	78	80.32	75.91	80.32
MLP_Mixer +TL	80.75	80.81	78.77	80.81
Proposed (Encoder: ResNet50)	**90.12**	**89.1**	**89.12**	**89.1**
Proposed (Encoder: VGG11)	**89.97**	**87.32**	**87.33**	**87.32**

Table 30 Validation results Dataset2. Dataset2 consists of 2-fold cross-validation. The best results highlighted.

Models (Avg. Dataset2)	Precision	Recall	F1-score	Accuracy
InceptionV3 [119]	76.88	75.44	74.45	75.44
VGG11 [110]	71.83	69.66	68.8	69.66
DenseNet [54]	72.4	70.61	69.2	70.61
ViT [117]	74.94	75.01	73.98	75.01
ResNet50 [7]	67.95	72.32	68.93	72.32
MLP_Mixer [118]	69.2	65.93	66.38	65.93
InceptionV3 +TL	72.12	67.28	62.42	67.28
VGG11 +TL	81.17	81.76	80.62	81.76
DenseNet +TL	78.72	77.52	76.61	77.52
ViT +TL	83.92	81.62	**82.11**	81.62
ResNet50 +TL	78.54	77.6	76.08	77.98
MLP_Mixer +TL	80.05	75.01	72.53	75.01
Proposed (Encoder: ResNet50)	**84.31**	**82.75**	82.09	**82.75**
Proposed (Encoder: VGG11)	**90.93**	**88.3**	**87.51**	**88.3**

Table 31 Validation results Dataset3. Dataset3 consists of 2-fold cross-validation. The best results highlighted.

Models (Avg. Dataset3)	Precision	Recall	F1-score	Accuracy
InceptionV3 [119]	73.64	72.39	70.07	72.39
VGG11 [110]	76.64	69.86	69.21	69.86
DenseNet [54]	66.29	66.15	64.73	66.15
ViT [117]	69.55	68.31	68.11	68.31
ResNet50 [7]	68.15	68.44	67.55	68.44
MLP_Mixer [118]	65.32	64.56	63.89	64.56
InceptionV3 +TL	80.77	77.1	74.58	77.1
VGG11 +TL	83.4	82.99	81.22	82.99
DenseNet +TL	82.73	81.87	80.2	81.87
ViT +TL	86.07	84.04	83.31	84.04
ResNet50 +TL	81.38	79.75	77.08	79.75
MLP_Mixer +TL	85.19	82.9	81.86	82.9
Proposed (Encoder: ResNet50)	**89.41**	**86.99**	**86.99**	**86.99**
Proposed (Encoder: VGG11)	**94.32**	**93.42**	**93.39**	**93.42**

Appendix C

To analyse the effect of normalising the vectors **r** and **g** in Equation (16) and Equation (17), the gradients of $\boldsymbol{\theta}$ needed first to be estimated. The gradients of $\boldsymbol{\theta}$ is given by the following equation (for more context, see Equation (23) in section 3.3.6):

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = \underbrace{\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}}}_{\text{Appendix B.2}} \underbrace{\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}}_{\text{Appendix B.1}} \tag{57}$$

In this appendix, the calculation of the gradients will be presented in two stages. The first stage will focus on the gradients of the input feature vector (**q**) with respect to the parameters of the fully-connected layer $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}$, which is derived in Appendix C.1. On the other hand, Appendix C.2 derives the gradients of the resultant feature vector of the fully-connected layer with respect to the SoftMax function and the Cross Entropy loss $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}}$.

To provide a contextual background for the calculations in Appendix C.1 and Appendix C.2, Figure 21 is presented once again below, as seen in Figure 46. Figure 46 illustrates the final layers of the proposed model, helping to establish a visual reference for understanding the calculations detailed in Appendix C.1 and Appendix C.2.

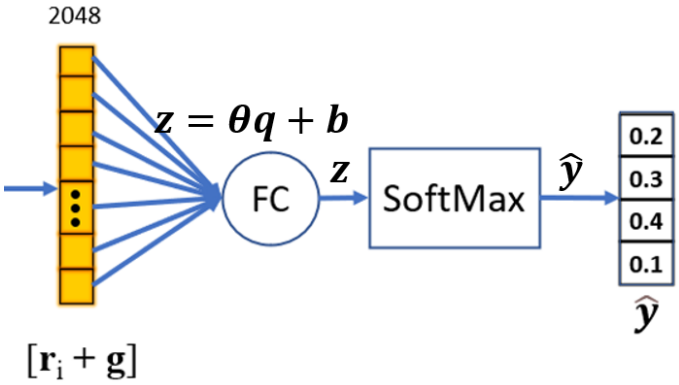


Figure 46 depicts the last fully connected layer that is responsible for generating probabilities vector. The full model architecture is depicted in Figure 12.

Appendix C.1

Given the gradients of $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}}$ (Appendix C.2), the gradients of $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}$ is defined as follows:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = (\hat{\boldsymbol{y}} - \boldsymbol{y})^T \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}, \in \mathbb{R}^{1\times(4\times 2048)} \tag{58}$$

The function **z** is an affine transformation that maps a high dimensional input vector to a much lower space:

$$\boldsymbol{z}: \mathbb{R}^{2048} \rightarrow \mathbb{R}^{n} \tag{59}$$

where n=4 represents the number of classes. Therefore, the gradients $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}$ is a Jacobian matrix of size $\mathbb{R}^{4\times(4\times2048)}$ [187]. Nevertheless, most of the elements of this Jacobian matrix are zeros as shown below:

$$\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}} = \begin{bmatrix} \frac{\partial z_1}{\partial \boldsymbol{\theta}} \\ \vdots \\ \frac{\partial z_n}{\partial \boldsymbol{\theta}} \end{bmatrix}, \quad \frac{\partial z_i}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{1\times(4\times2048)} \tag{60}$$

$$z_i = \sum_{j=1}^{2048} \theta_{ij} \cdot q_j + b_i\,, \qquad i = 1, \dots, n \tag{61}$$

The partial derivatives w.r.t θ_{ij} are then given as follows:

$$\frac{\partial z_i}{\partial \theta_{ij}} = q_j \tag{62}$$

Hence the derivatives w.r.t the row $\theta_{i,:}$ are given as follows:

$$\frac{\partial z_i}{\partial \theta_{i,:}} = \boldsymbol{q}^T \in \mathbb{R}^{1\times1\times2048} \tag{63}$$

$$\frac{\partial z_i}{\partial \theta_{w\neq i,:}} = \boldsymbol{0}^T \in \mathbb{R}^{1\times1\times2048} \tag{64}$$

Thus, the partial derivatives $\frac{\partial z_i}{\partial \boldsymbol{\theta}}$ is given by:

$$\frac{\partial z_i}{\partial \boldsymbol{\theta}} = \begin{bmatrix} \boldsymbol{0}^T \\ \vdots \\ \boldsymbol{q}^T \\ \vdots \\ \boldsymbol{0}^T \end{bmatrix} \in \mathbb{R}^{1\times(4\times2048)} \tag{65}$$

By ignoring the zeros in Equations (65), the gradients $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}} \in \mathbb{R}^{4\times(4\times2048)}$ boils down to a Jacobian matrix filled mostly with $\boldsymbol{0}^T$ and four $\boldsymbol{q}^T$ vectors.

Appendix C.2

In this section, the gradients of $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}}$ will be derived to facilitate the calculation of the gradients of $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}$. The derivation presented here utilised in Appendix C.1 and section 3.3.6. First the gradients $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \in \mathbb{R}^{1\times4}$ will be calculated, followed by the gradients $\frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}} \in \mathbb{R}^{4\times4}$, and then multiply the two gradients to obtain $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}}$. However, leveraging the fact that the logarithm function in the Cross Entropy loss $\mathcal{L}$ (i.e., Equation (21)) is the inverse function of the exponential functions

used in the SoftMax function (i.e., Equation (20)), which simplifies the derivation process. Therefore, the composite function $(\mathcal{L} \circ SoftMax)(\boldsymbol{z}) \in \mathbb{R}^4$ will be calculated first, and then derive the gradients accordingly:

$$(\mathcal{L} \circ SoftMax)(\boldsymbol{z}) = -\sum_{i=1}^{4} y_i \log\big(SoftMax(z_i)\big)$$

$$= -\sum_{i=1}^{4} y_i \log\left(\frac{e^{z_i}}{\sum_{l=1}^{n} e^{z_l}}\right) \tag{66}$$

$$= -\sum_{i=1}^{4} y_i log(e^{z_i}) - y_i \log\left(\sum_{l=1}^{4} e^{z_l}\right) = \sum_{i=1}^{4} -y_i z_{\boldsymbol{i}} + y_i \log\left(\sum_{l=1}^{4} e^{z_l}\right) \tag{67}$$

Using vector compact notation and given that $\boldsymbol{y}$ is a one-hot vector (e.g., $\boldsymbol{y} = [0100]^T$), Equation (67) can be written as follows:

$$-\boldsymbol{y}^T\boldsymbol{z} + \log\left(\sum_{l=1}^{n} e^{z_l}\right) \cdot \sum_{i=1}^{4} y_i = -\boldsymbol{y}^T\boldsymbol{z} + \log\left(\sum_{l=1}^{n} e^{z_l}\right) \cdot 1 \tag{68}$$

$$\therefore (\mathcal{L} \circ SoftMax)(\boldsymbol{z}) = log\left(\sum_{\boldsymbol{l=1}}^{\boldsymbol{n}} e^{z_l}\right) - \boldsymbol{y^T z} \tag{69}$$

The gradients of the composite function $\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}}$ can be calculated directly by using Equation (69):

$$\frac{\partial \mathcal{L}}{\partial \hat{\boldsymbol{y}}} \frac{\partial \hat{\boldsymbol{y}}}{\partial \boldsymbol{z}} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{z}} = \frac{\partial}{\partial \boldsymbol{z}} log\left(\sum_{\boldsymbol{l=1}}^{\boldsymbol{n}} \boldsymbol{e^{z_l}}\right) - \frac{\partial}{\partial \boldsymbol{z}} \boldsymbol{y^T z} = \left(\frac{\boldsymbol{e^z}}{\sum_{\boldsymbol{l=1}}^{\boldsymbol{n}} \boldsymbol{e^{z_l}}}\right)^T - \boldsymbol{y}^T \tag{70}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{z}} = \hat{\boldsymbol{y}}^T - \boldsymbol{y}^T = (\hat{\boldsymbol{y}} - \boldsymbol{y})^T \in \mathbb{R}^{1\times 4} \tag{71}$$

The gradients of $\frac{\partial \mathcal{L}}{\partial \boldsymbol{z}}$ are used in section 3.3.6 to derive the gradients of the fully-connected layer $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}$ and accordingly analyse the effects of normalising the feature vectors **r** and **g**.

Appendix D

This appendix presents an experiment related to the proposed image-to-image transformation, namely, Total Variational $TV_{\emptyset}$ and Texture Interpolation $TI_{\emptyset}$, explained in Chapter 4. The experiment has no direct effects on the downstream task (i.e., polyp segmentation), hence they are deferred here for discussion. Appendix D.1 articulate checkerboards emerged due to selecting different gradient filter for calculating total variation loss $\|\nabla\check{x}\|$. Meanwhile, Appendix D.2 provides additional demonstration of $TV_{\emptyset}$ and $TI_{\emptyset}$ transformations during the training phase.

Appendix D.1

The objective of Total Variational $TV_{\emptyset}$ is to minimize the total variations of the background, meanwhile retains polyp texture as follows:

$$\min_{\check{x}} \begin{cases} \alpha\|x(w,h) - \check{x}(w,h)\|^2 & for \quad (w,h) \in \boldsymbol{A} \\ \beta\|\nabla\check{x}\| & otherwise \end{cases} \tag{72}$$

where A is a set of points that represent coordinates of polyp pixels. (w, h) represents pixel coordinates, whereby, α and β are scalars used to balance between construction and total variation $\|\nabla\check{x}\|$ objectives. The Sobel derivative filter was utilised to calculate $\nabla\check{x}$ and accordingly checkerboard artefacts were noticed as shown in Figure 47 at first row.

However, this effect vanishes when central difference is used as seen in the third row in Figure 47. Nevertheless, regardless of the used derivative, a segmentation model would learn to ignore this artefact during the training phase. Accordingly, utilising any derivative operator will not affect the downstream task.

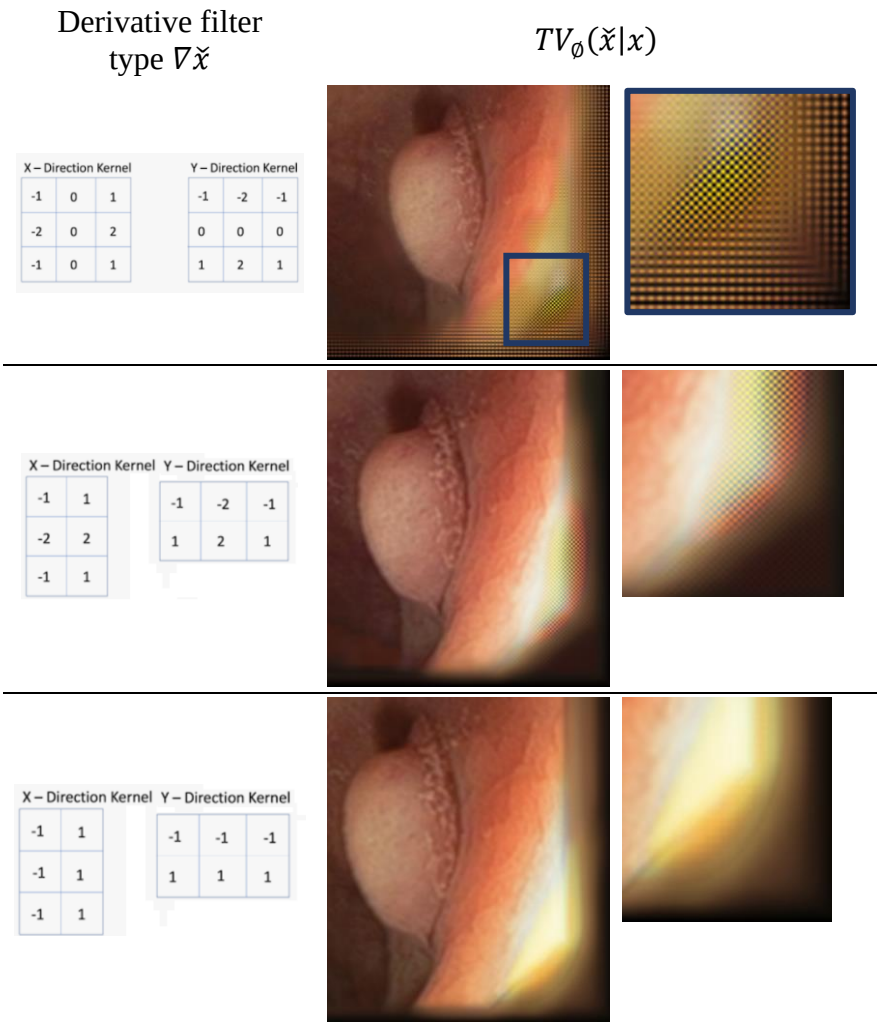


Figure 47 The effect of different derivative operators on the transformed image $\check{x}$.

Appendix D.2

As illustrated in Chapter 4, the Texture Variational $TV_{\emptyset}$, which is a deep learning unit, applies online texture transformations to input images in order to provide a segmentation model with various transformed images during the training phase. The transformation process decreases image's gradient of the background meanwhile reserve polyp regions. Accordingly, the segmentation model gets exposed to various vanishing texture levels, hence, it obtains texture invariance properties. Figure 48 and Figure 49 show transformation progress along with the generated mask by the segmentation model. At the first epoch both the $TV_{\emptyset}$ and the segmentation model produces unsatisfactory results, though, they gradually learn to achieve their corresponding objectives as training epochs progress.

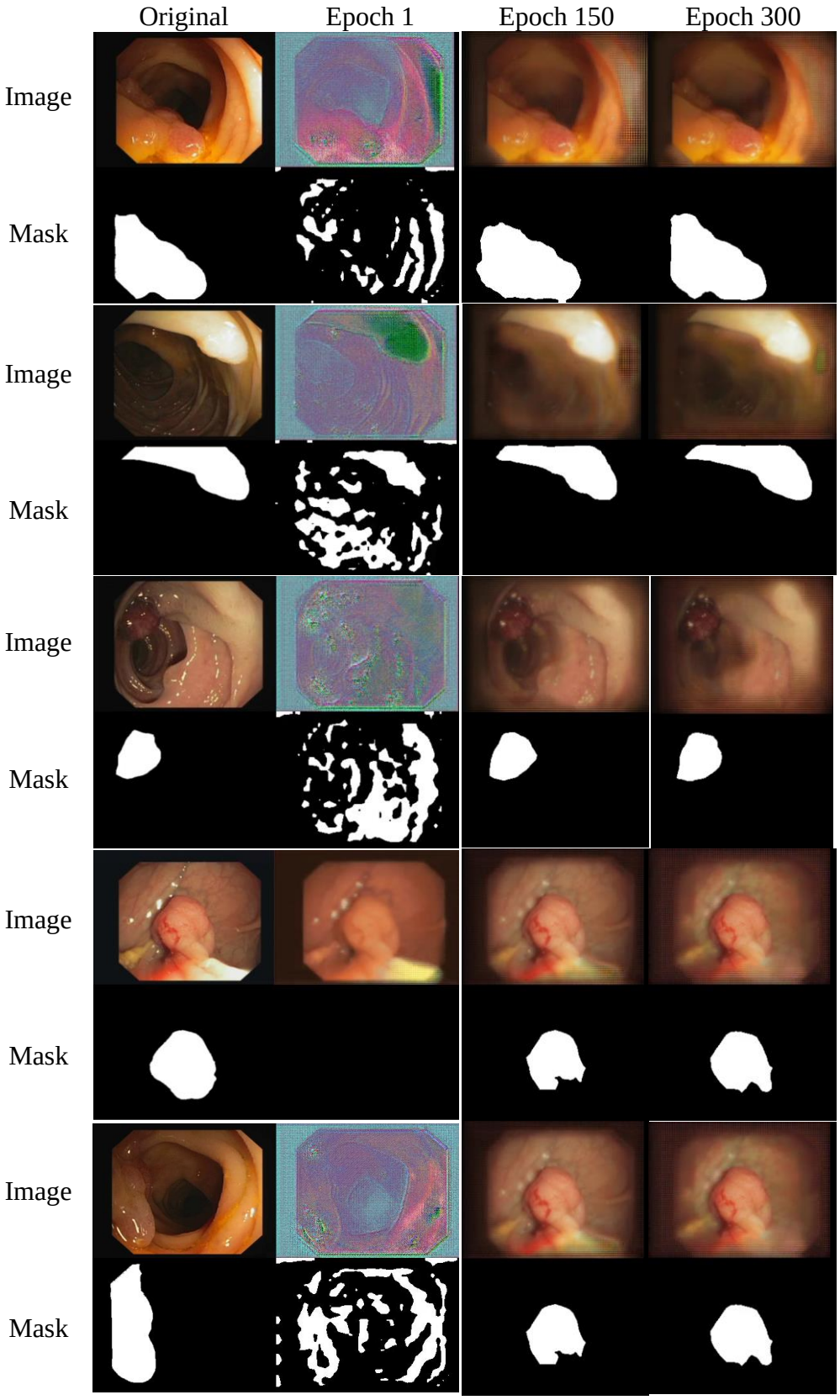


Figure 48 $TV_{\emptyset}$ transformations and mask generation during training phase.

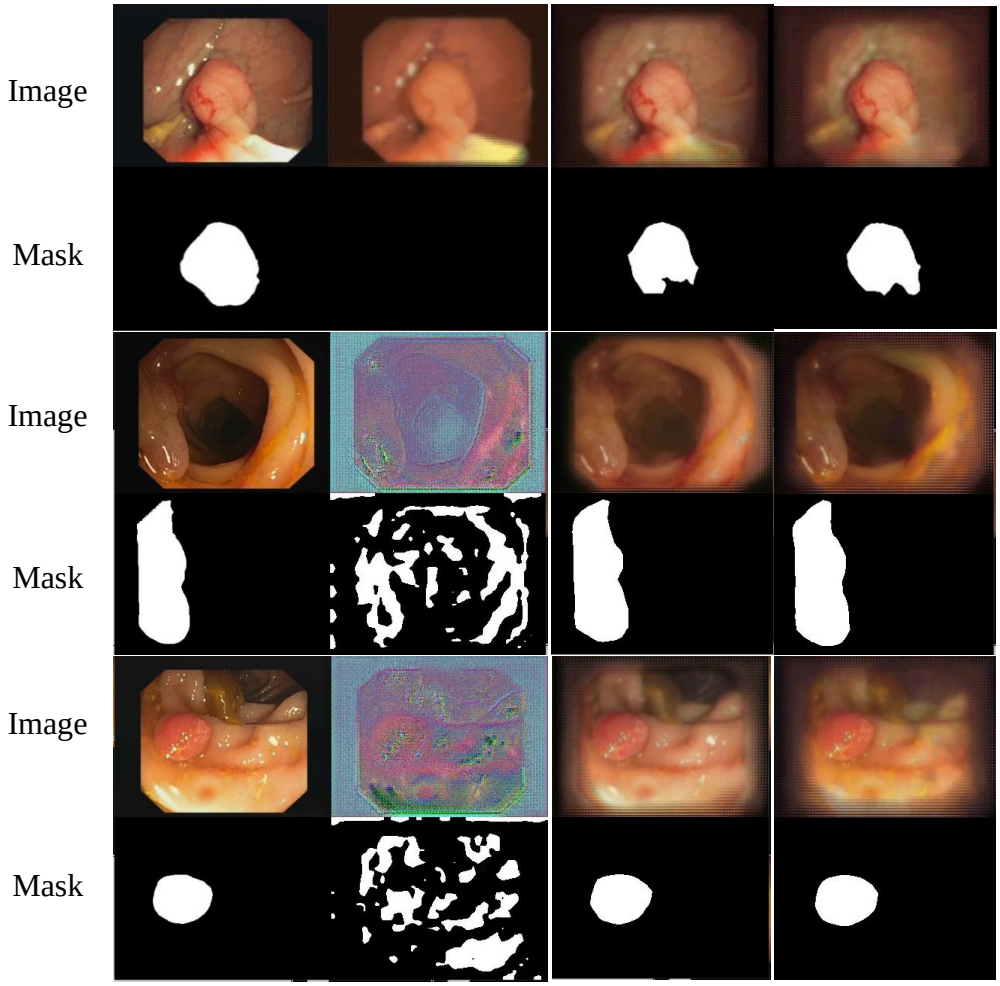


Figure 49 $TV_{\emptyset}$ transformations and mask generation during training phase. More examples.

As discussed in Chapter 4, Texture Interpolation $TI_{\emptyset}$ unit is another implementation of the proposed framework. The $TI_{\emptyset}$ produce an interpolated image given an original image and a textureless image produced by a pre-trained unit (i.e., Autoencoder unit). Therefore, $TI_{\emptyset}$ produces valid images starting from Epoch 1 as opposed to the $TV_{\emptyset}$, as seen in Figure 50. However, the segmentation unit gradually learn to segment polyp regions as the training progresses.

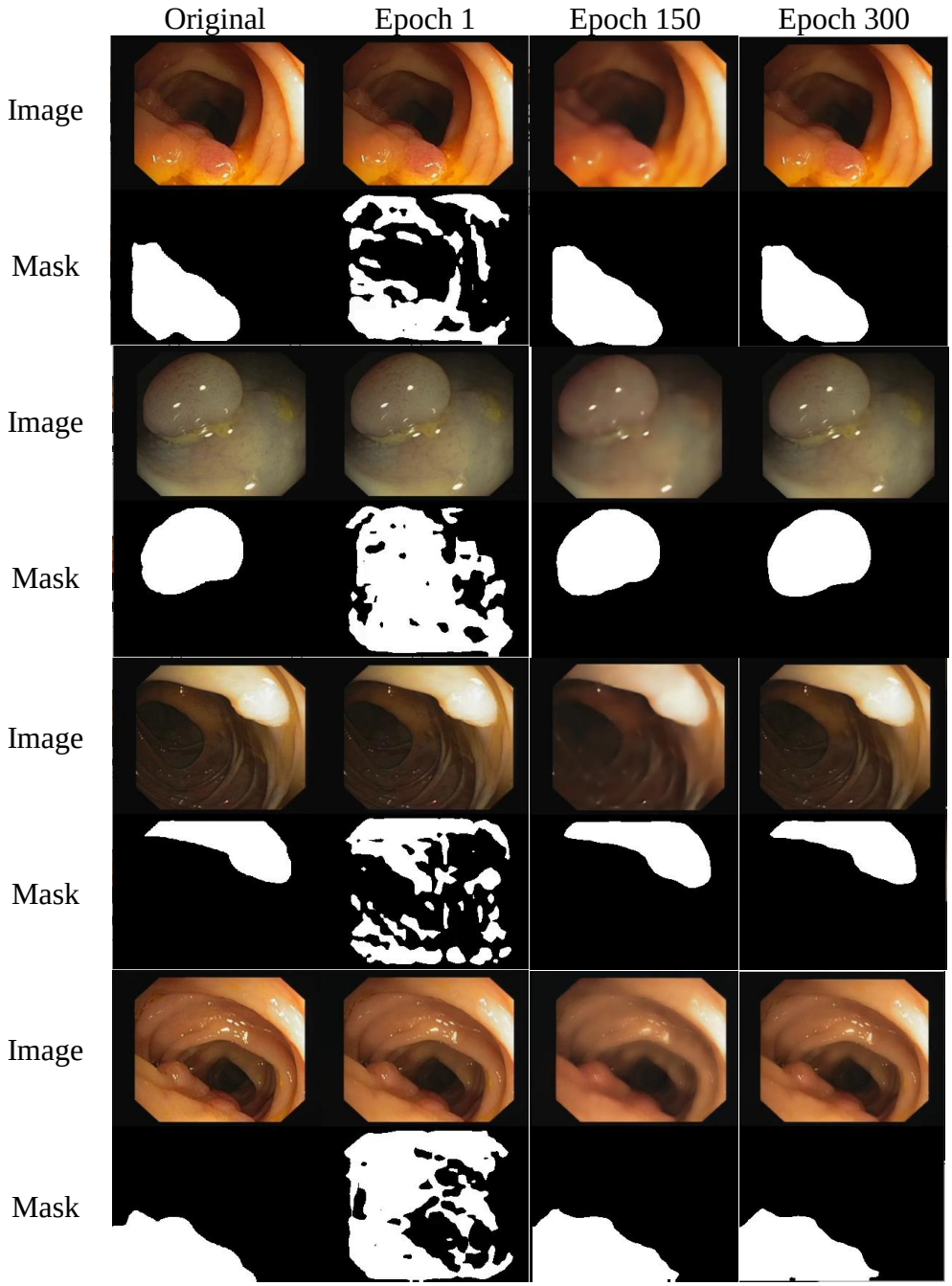


Figure 50 $TI_{\emptyset}$ transformations and mask generation during training phase.

Figure 51 shows the progress of the Autoencoder AE component that is used in the $TI_{\emptyset}$ unit. The AE unit was assigned a polyp mask generation task along with input reconstruction task to prevent it from learning identity function. Accordingly, AE learns to reconstruct original images without texture details, as seen in Figure 51. As it can be seen from Figure 51, AE failed to reconstruct original images at the first training epoch, however after a while, it learns to reconstruct main features of input images.

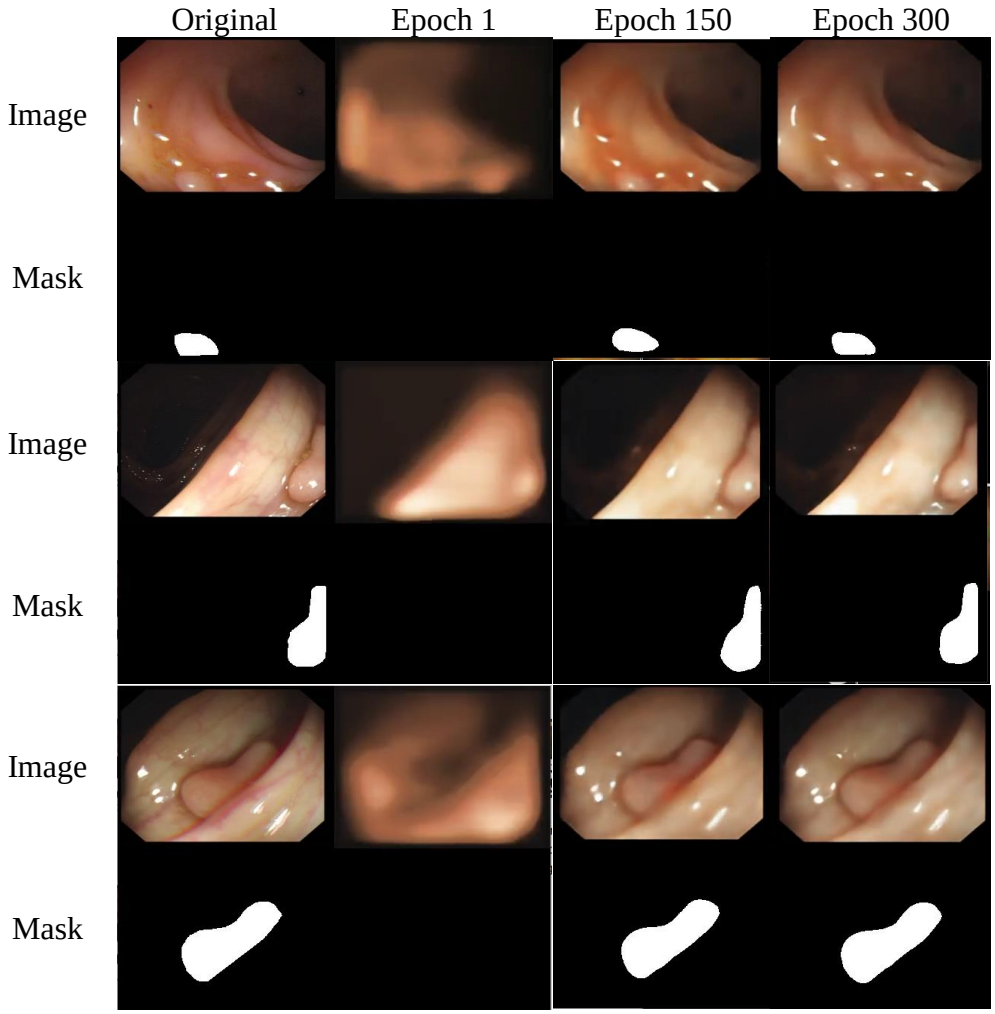


Figure 51 Training the Autoencoder in $TI_{\emptyset}$ unit to reconstruct original image.