A Comparative Study on Polyp Classification and Localization from Colonoscopy Videos

©2019

Mohammad Isyroqi Fathan

B.S. Computer Science, University of Kansas, 2017

Submitted to the graduate degree program in Department of Electrical Engineering and Computer Science and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Master in Computer Science.

Committee members

Dr. Guanghui Wang, Chairperson

Dr. James Miller, Member

Dr. Bo Luo, Member

Date defended: May 28, 2019

The Thesis Committee for Mohammad Isyroqi Fathan certifies
that this is the approved version of the following thesis :

A Comparative Study on Polyp Classification and Localization from Colonoscopy Videos

Dr. Guanghui Wang, Chairperson

Date approved: June 06, 2019

Abstract

Colorectal cancer is one of the most common types of cancer with a high mortality rate. It typically develops from small clumps of benign cells called polyp. The adenomatous polyp has a higher chance of developing into cancer compared to the hyperplastic polyp. Colonoscopy is the preferred procedure for colorectal cancer screening and to minimize its risk by performing a biopsy on found polyps. Thus, a good polyp detection model can assist physicians and increase the effectiveness of colonoscopy. Several models using handcrafted features and deep learning approaches have been proposed for the polyp detection task.

In this study, we compare the performances of the previous state-of-the-art general object detection models for polyp detection and classification (into adenomatous and hyperplastic class). Specifically, we compare the performances of FasterRCNN, SSD, YOLOv3, RefineDet, RetinaNet, and FasterRCNN with DetNet backbone. This comparative study serves as an initial analysis of the effectiveness of these models and to choose a base model that we will improve further for polyp detection.

Acknowledgements

I would like to thank my advisor Dr. Richard Wang who has given me help and guidance throughout my Master's program at the University of Kansas. Dr. Richard Wang has been a passionate and understanding advisor who always try to push his students to improve on their skills in research as well as professional development. I would also like to thank Dr. James Miller and Dr. Bo Luo for their help with my thesis, defense, and also for their enriching courses that I have taken during my Master program. The amazing experience I have had at the University of Kansas would not be possible without the hard work of all the University of Kansas faculty and staff members. I am really grateful for the supportive environment towards my professional development that I have had during my study at the University of Kansas.

I would also like to thank my parents and my family for their continuous support for me. I would not have done it without their support during my ups and downs. I am also thankful for all my friends here at the University of Kansas who have been helpful towards me and who have taught me so much about life. Finally, I am also thankful for the students in Dr. Wang's Computer Vision group who have been very helpful in enriching my knowledge throughout deep technical discussions about computer vision field that we are passionate about.

Contents

1 Introduction 1

1.1 Motivation 1

1.1.1 Colorectal Cancer 1

1.1.2 Motivations and Goals 3

1.1.3 Challenges and Problems 4

1.2 Contributions 6

1.3 Outline 6

2 Literature Review 8

2.1 Computer Vision Problem Statements 8

2.2 Handcrafted Features Approach and Deep Learning Approach 9

2.3 Deep Learning Object Detection Categories 11

2.3.1 Two-stage Frameworks 11

2.3.2 One-stage Frameworks 13

3 General Object Detection Model 15

3.1 Faster RCNN 15

3.2 YOLOv3 17

3.3 Single Shot Detector 18

3.4 RetinaNet 21

3.5 RefineDet 22

3.6 DetNet 25

4 Experimentation 27

4.1 Dataset 27

4.1.1 MICCAI 2017: GIANA Endoscopic Vision Challenge Dataset 27

4.1.2 CVC-ColonDB Dataset 27

4.1.3 Gastrointestinal Lesions in Regular Colonoscopy Dataset 28

4.1.4 KUMC 80 Videos Dataset 28

4.2 Dataset Preparation 28

4.2.1 Ground Truth Bounding Box Annotation 29

4.2.2 Frame Selection 29

4.2.3 Dataset Split 30

4.2.4 Dataset Summary 31

4.3 Experiment Settings 31

4.3.1 Model Hyperparameter Settings 32

4.4 Comparison Methods 33

4.4.1 Polyp Detection 33

4.4.2 Polyp Localization 34

4.4.3 1 Class Polyp Classification VS 2-Classes Polyp Classification 34

4.4.4 Still Frame vs Video Sequence 35

4.4.5 Combined Performance 35

4.4.6 Comparison Metrics 35

5 Results and Discussions 38

5.1 Confidence Score Analysis 40

5.2 1-class Polyp Image Classification Task 43

5.3 2-classes Polyp Image Classification Task 46

5.4 1-class Polyp Localization Task 49

5.5 2-classes Polyp Localization Task 51

5.6 Video Sequence Classification Task 54

5.7 Image Frame Prediction Analysis 59
5.8 Video Sequence Analysis 68
5.9 Image Frame Prediction Summary 71

6 Conclusions **80**

7 Future Works **82**

List of Figures

1.1 Paris Classification of Superficial Neoplastic Lesions (Type 0) 5

2.1 Various Computer Vision Tasks Applied to Colonoscopy 10

3.1 FasterRCNN model architecture. Source: Ren et al. (2015) 16
3.2 SSD architecture and anchor box. Source: Liu et al. (2016) 20
3.3 RetinaNet architecture. Source: Lin et al. (2017) 22
3.4 RefineDet architecture and Transfer Connection Block Module. Source: Zhang et al. (2018) 24
3.5 DetNet architecture. Source: Li et al. (2018) 26
3.6 DetNet architecture. Source: Li et al. (2018) 26

4.1 Summary of datasets 31
4.2 Visualization of dataset split 31

5.0 Distribution of prediction confidence scores for each model, green shows correct prediction, while red shows false positive prediction 41
5.1 ROC Curve for 1-class polyp detection 44
5.2 PR Curve for 1-class polyp detection 44
5.3 ROC Curve for 2-classes polyp detection: adenomatous polyp 47
5.4 ROC Curve for 2-classes polyp detection: hyperplastic polyp 47
5.5 PR Curve for 2-classes polyp detection: adenomatous polyp 48
5.6 PR Curve for 2-classes polyp detection: hyperplastic polyp 48
5.7 PR Curve for 1-class polyp localization 50
5.8 PR Curve for 2-classes polyp localization: adenomatous polyp 52

5.9 PR Curve for 2-classes polyp localization: hyperplastic polyp 52
5.10 ROC Curve for 2-classes polyp detection per video sequence: adenomatous polyp 55
5.11 ROC Curve for 2-classes polyp detection per video sequence: hyperplastic polyp . 55
5.12 PR Curve for 2-classes polyp detection per video sequence: adenomatous polyp . . 56
5.13 PR Curve for 2-classes polyp detection per video sequence: hyperplastic polyp . . 56
5.14 Video sequence class prediction probability 57
5.15 Percentage of total correct prediction for each model 60
5.16 Percentage of total misclassified prediction for each model 61
5.17 Percentage of total mislocalized prediction for each model 62
5.18 Percentage of total multiple predictions for each model 63
5.19 Percentage of total polyp frames without prediction for each model 65
5.20 Percentage of total non polyp frames with prediction for each model 66
5.19 Percentage of correctly predicted frames per video sequence for each model. Numbers on top of the bars show the percentage values, while numbers below the bars show the actual number of frames 70

List of Tables

5.1 Performance of each model under different metrics for 1-class polyp detection task (classifying each frame as polyp or non-polyp) 43

5.2 Performance of each model under different metrics for 2-classes polyp detection task (classifying each frame as adenomatous, hyperplastic, or non-polyp) 46

5.3 Performance of each model under different metrics for 1-class polyp localization task (locating and classifying each prediction as polyp or non-polyp) 49

5.4 Performance of each model under different metrics for 2-classes polyp localization task (locating and classifying each prediction as adenomatous, hyperplastic, or non-polyp) 51

5.5 Performance of each model under different metrics for video sequence classification task (classifying each video sequence as adenomatous, hyperplastic, or non-polyp) 54

5.6 Summary for number of frames and classification of each video sequence 73

5.7 Number of correctly predicted frames. Here, correct prediction means correct classification, localization, and number of prediction each frame 74

5.8 Number of misclassified prediction frames. Here, misclassified prediction means frame containing prediction at the correct location but wrong classification prediction 75

5.9 Number of misslocalized prediction frames. Here, misslocalized prediction means frame containing prediction at the wrong location (regardless of class prediction) . 76

5.10 Number of multiple predictions frames. Here, multiple predictions means frame containing more than one predictions 77

5.11 Number of polyp frames without prediction. Here, it means frame containing polyp but the model predicted that it does not contain polyp 78
5.12 Number of non-polyp frames with prediction. Here, it means frame containing no polyp but the model predicted that it contains polyp 79

Chapter 1

Introduction

1.1 Motivation

1.1.1 Colorectal Cancer

Colorectal cancer is one of the most common types of cancer with a high mortality rate. According to studies, colorectal cancer is the fourth highest cancer by rates of new cancer cases and also the fourth highest cancer by rates of deaths as of 2015 Group (2018); Society (2019b). Furthermore, American Cancer Society estimated that colorectal cancer will be the fourth highest new cancer cases and the second highest cause of deaths by cancer in 2019 Society (2019b). This shows us the importance of developing accurate early screening methods as well as treatment techniques for colorectal cancer.

Colorectal cancers typically develop from small clumps of benign cells called polyps Simon (2016). Colon polyps can potentially grow slowly to become colorectal cancer over a period of 10 to 20 years. Due to this slow growth, an early screening process to detect the presence of these polyps can help prevent their potential future growth to become cancerous cells. Screening can reduce the incidence of disease and increase the likelihood of survival Society (2019a). Increasing age is one of the greatest risk factors to colorectal cancer, with 99% of cases occurred in people of age more than 40 and 85% in those of age more than 60 Ballinger & Anggiansah (2007). Thus, it is recommended to get colorectal screening beginning at the age of 50 (although, family history can increase the risk even for those aged below 50).

Common CRC screening methods can be categorized into two; visual examinations and stool-based tests. In stool-based tests, the stool (feces) are checked for signs of cancer such as blood (guaiac-based fecal occult blood and fecal immunochemical tests), and additionally for genetic mutation in the DNA of cells that are shed into the stool (FIT-DNA/Cologuard test). While having the benefits of being non-invasive, no bowel cleansing necessary, and can be performed at home, these tests often miss most polyps and require shorter test time interval.

Colonoscopy is the recommended visual examination screening method, although there are other alternatives. CT colonography uses X-ray to get 2D or 3D views of the entire colon and rectum. While being less invasive, it suffers from low detection rate on smaller polyps (5mm or less) Johnson et al. (2008); de Haan et al. (2011). Similarly, double-contrast barium enema also uses X-ray, but it has a lower detection rate than CT colonography Johnson et al. (2004). Sigmoidoscopy requires less complicated bowel preparation, but it only covers the rectum and lower third of the colon. Wireless Capsule Endoscopy has high sensitivity for polyp detection, but it suffers greatly from its dependence on whether the polyp is recorded during its progression through the gastrointestinal tract and from the length of the recorded video (around 8 hours) Wang et al. (2013); Spada et al. (2011); Eliakim et al. (2009). This long analysis is highly time consuming and can suffer from physician's fatigue, affecting the polyp detection rate. Nevertheless, any positive tests (either stool-based or visual examination test) require further colonoscopy for complete diagnostic, making colonoscopy an important screening method.

Colonoscopy procedure allows the physician to examine the entire colon and perform biopsies on detected polyps. Colonoscopy requires a good bowel preparation, which affects the polyp detection rate Rees et al. (2016); Lebwohl et al. (2011). Depending on the number and size of polyps found, physicians may need to operate the colonoscope for a long time, which may increase polyp miss detection rate due to mental and physical fatigue. Furthermore, a study showed that colonoscopy procedure has a 25% miss rate for all polyps Leufkens et al. (2012). Therefore, automatic computer aided-system is needed to improve the effectiveness of colonoscopy.

Colorectal polyps are commonly divided into two categories, non-neoplastic (commonly hyperplastic) and neoplastic (commonly adenomatous) polyps Shinya & Wolff (1979). Hyperplastic polyps are commonly serrated, diminutive ($\leq$ 5mm), pliable, and sessile KIM & PICKHARDT (2010). Although it is often considered to have little to no malignant potential Roland & Barnett (2009), hyperplastic polyps should not be ignored Jass (2004). Adenomatous polyps are the more common precursors as they account for approximately 85% of sporadic colorectal cancers, termed the adenoma-carcinoma pathway KIM & PICKHARDT (2010). Thus, detected adenomatous polyps are often removed during colonoscopy procedures.

1.1.2 Motivations and Goals

It has become apparent that a more accurate and effective automatic computer-aided system for colonoscopy is needed to help physicians detect possible precancerous colorectal polyps as early as possible. Computer-aided diagnosis may help physicians to avoid missing polyps and misdiagnosing their types, especially due to mental and physical fatigues.

Common computer-aided systems developed for polyp detection in colonoscopy video approaches are utilizing handcrafted features and classical machine learning approach to locate and classify the polyp. Often time, this approach can result in poor performance when there is a slight variation in the frame which causes the feature to be unreliable. Deep learning has been getting more popular in the computer vision field due to its success in solving numerous problems in computer vision.

Here, we are particularly interested in classical general object detection models using deep learning approach such as FasterRCNN Ren et al. (2015), YOLOv3 Redmon & Farhadi (2018), SSD Liu et al. (2016), RetinaNet Lin et al. (2017), RefineDet Zhang et al. (2018), and DetNet Li et al. (2018). The goal is to compare these classical general object detection models for polyp detection and classification in colonoscopy as baseline models that could potentially be improved to be better suited for polyp detection. While similar study exists Bernal et al. (2017), this study compares proposed models for MICCAI 2015 Challenge on Automated Polyp Detection. In this challenge, the goal was to locate and detect polyps either from still frames or sequence of video frames. Note that this challenge does not classify the polyps to either adenomatous or hyperplastic, as opposed to our study. Furthermore, some of the proposed models use handcrafted features, hybrid, or end-to-end deep learning approach.

1.1.3 Challenges and Problems

When developing a computer-aided system for polyp detection in colonoscopy, we need to address some possible challenges and problems. Being in a medical field, the small availability of data (compared to popular large object detection datasets like ImageNet Deng et al. (2009) and PascalVOC Everingham et al. (2010a)) is often a problem for deep learning approach as the success of deep learning often depends on the size and quality of the dataset Razzak et al. (2018). The practical usefulness of the model also depends on the real-time capability of the proposed model, since colonoscopy procedures are performed in real-time. Furthermore, some challenges may arise from the hardware perspective and from the scene environment perspective.

From the hardware perspective, various advancements in colonoscopy technology have been developed to help increase physicians detection accuracy Ngu & Rees (2018). The use of high-definition colonoscopy increases the resolution and quality of image, thus increases textural information in the image. The use of zooming and magnification technology, as well as wide-angle camera, might also help to capture more of the colon surface. Conventional chromo-endoscopy uses contrast dyes to enhance the characterization of tissues, mucosal surfaces, and blood vessels. Virtual chromo-endoscopy also tries to achieve the same goal using a narrow spectrum of wavelengths with a decreased penetration depth Ngu & Rees (2018), as opposed to conventional white-light endoscopy. Variants of virtual chromo-endoscopy include Narrow-band Imaging, Fuji Intelligent Color Enhancement, Autofluorescence Imaging, i-SCAN, and Endoscopic Trimodal Imaging. All these various technologies developed and used in colonoscopy must be taken into consideration when building a computer-aided system to gain advantages of each technology as much as possible.

From the scene environment perspective, we need to take into account of other textures that might be present in the colon. Since colonoscopy requires a good bowel preparation Bechtold et al.

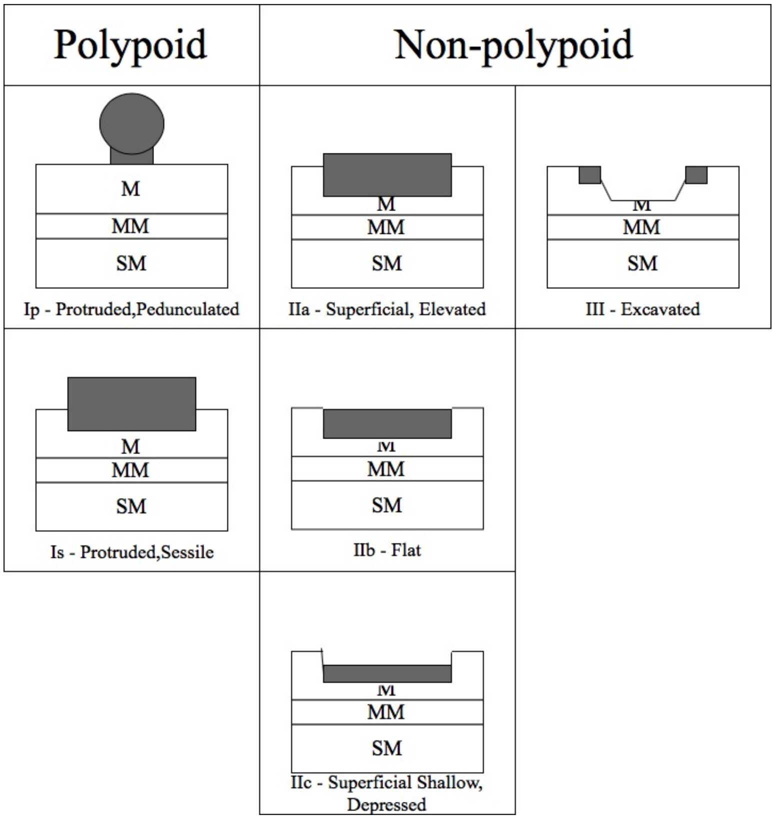


Figure 1.1: Paris Classification of Superficial Neoplastic Lesions (Type 0)

(2016); Saltzman et al. (2015), the model needs to be robust to possible presences of solid/semi-solid stools as well as moderate to large amount of liquid and fluid. Other than residual particles and liquid from bowel preparation, blood vessels and colon wall textures might affect the appearance of polyps, and thus, the detection of the polyps. Another factor that contributes more towards polyp detection is the polyp morphology. Paris classification Workshop (2003) categorizes superficial neoplastic lesions (type 0) into two subtypes, which are polypoid and non-polypoid subtypes. Major variants of superficial neoplastic lesions are shown in Fig. 1.1. Combinations of the variants (such as IIa + Is and IIa + IIc) are also possible. A robust model has to be able to detect these different variants of polyp morphology. Sessile adenomatous polyps are known to have higher detection miss rate than pedunculated adenomatous polyps Kim et al. (2017). Furthermore, it has also been known that polyp detection miss rate increases with smaller polyp size Ngu & Rees (2018); Van Rijn et al. (2006). So, the model must also be robust to variation of polyp sizes.

1.2 Contributions

To the best of our knowledge, this thesis contributes to the first comparative study of classical general object detection models as baseline models for automated polyp classification and localization. This thesis compares the performances of the object detection models using specified metrics and provides the baseline performances of these models.

This thesis also contributes an open source dataset for polyp classification (to adenomatous polyp or hyperplastic polyp) and localization, which we built by combining existing polyp detection datasets and a private colonoscopy video dataset.

1.3 Outline

The rest of this thesis is organized as follows. In chapter 2, we review different types of computer vision problem statements. Then, we briefly discuss some handcrafted features approaches in polyp detection and compare them with some proposed deep learning approaches. Then, we discuss the difference between two-stage framework and one-stage framework in object detection using deep learning.

In chapter 3, we summarize the model specifications of general object detection models that we compare in this thesis. We also provide overviews about the models we compare in this thesis.

In chapter 4, we present the details of our experiments. We start by describing the datasets that we use, the dataset preparation steps, and summary about the dataset. Then, we explain the experiment settings, specifically the hyperparameters we chose for training and evaluating the models. Finally, we explain the comparison methods and metrics used throughout our experiments.

In chapter 5, we present the experiment results and analysis. We compare and discuss our results based on our observations.

In chapter 6, we conclude our findings from the experiment. We found that RefineDet has the best performance compared to all the other models because it performs significantly better on hyperplastic frames. SSD has the best performance on adenomatous frames and has the second best performance after RefineDet.

In chapter 7, we propose some possible future works to improve the best performing model (RefineDet) for polyp classification and localization.

Chapter 2

Literature Review

2.1 Computer Vision Problem Statements

Computer vision is one of the fastest growing fields in research due to the invention of deep learning and convolutional neural network. Since AlexNet introduction and success in 2012 Krizhevsky et al. (2012) for ImageNet Large Scale Visual Recognition Challenge (ILSVRC), most computer vision researches have been focusing on deep learning methods.

Various computer vision tasks from the simplest to the more complex tasks are *image-level object classification*, *object detection*, *semantic image segmentation*, and *object instance segmentation* Liu et al. (2018). In *image-level object classification* task, the model is presented with an image and asked for probable classes of the objects contained in the image or asked about the presence of particular objects (classes) within the image, without the need of information regarding the locations of the objects. *Object detection* task expands this task by asking about the presence of objects within the image along with the locations and bounding box surrounding each detected object. This task is often expanded to detect multiple classes of objects within an image. *Semantic image segmentation* task focuses on pixel level prediction of the image. In this task, the model predicts the most probable class assignment for each particular pixel in the image. *Object instance segmentation* task is like the combination of object detection and semantic image segmentation in which it predicts the most probable class assignment of each pixel in the image while distinguishing different instances of the objects.

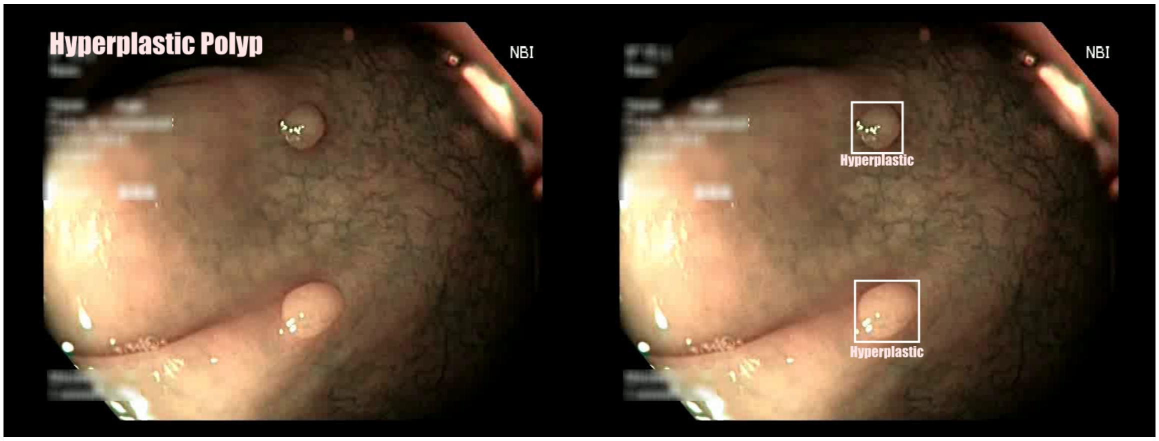
While the four tasks discussed previously often appear in medical image analysis, there are also other tasks that we may find Litjens et al. (2017). In *registration* task, the model tries to find the best alignment (commonly spatial alignment) of multiple medical images so that we can make a better comparison of the images from multiple patients. This is due to the shape and size variations between patients' bodies or organs. In *content-based image retrieval* task, the model extracts features from the image, which will later be used to compute distances between other images in the database and finally retrieve the closest matching images. *Image generation and enhancement* task tries to improve the quality of the image. Lastly, *combining image data with report* task tries to generate a caption or semantic label describing the image.

Fig. 2.1 depicts how each of the four main tasks would look like in computer-aided colonoscopy. Due to the nature of colonoscopy procedures, image-level object classification will not be sufficient in helping physicians to locate the polyps. Furthermore, the continuous nature of colonoscopy video image frames will be difficult for image level object classification models to output the correct prediction due to the variation of locations, sizes, and ratios of the polyps within the image. Semantic image segmentation and object instance segmentation which predict in pixel level might be very helpful for physicians to actually see the locations as well as the boundaries of the polyps. However, coarse bounding boxes of the polyps are often sufficient to help physicians locating the polyps found during colonoscopy procedures. Thus, we chose to do comparison of models for object detection and classification task in this study.

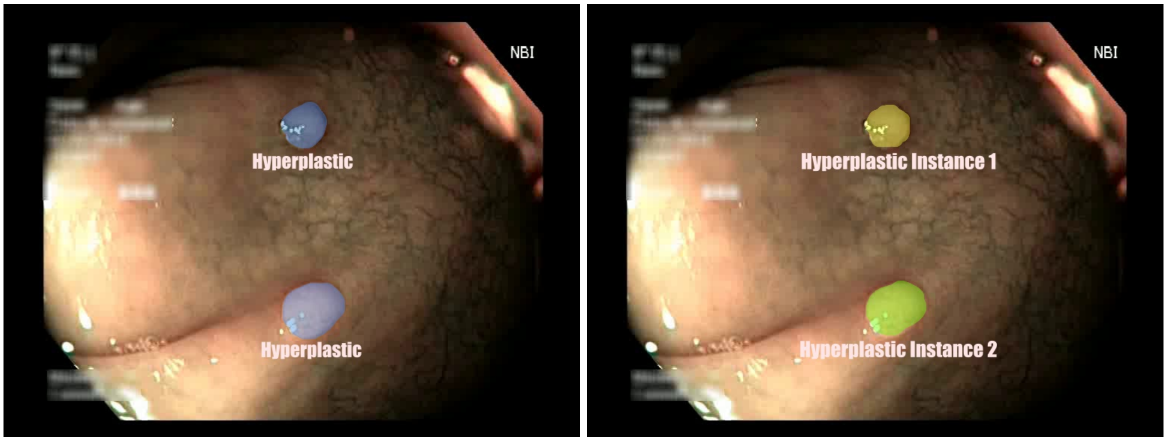
2.2 Handcrafted Features Approach and Deep Learning Approach

There have been various models proposed for polyp detection in colonoscopy. Previous comparative validation study on MICCAI 2015 Polyp Detection Challenge Bernal et al. (2017) includes previously proposed models, both using handcrafted feature approach and deep learning approach.

Handcrafted feature approaches focus on using low-level image processing methods to extract geometric shape features or texture description features. Handcrafted feature approaches often perform in real-time, making it suitable for real-life application. Furthermore, handcrafted



(a) Image Level Object Classification (b) Object Detection



(c) Semantic Image Segmentation (d) Object Instance Segmentation

Figure 2.1: Various Computer Vision Tasks Applied to Colonoscopy

feature approaches do not need a large amount of dataset compared to deep learning approaches. CVC-Clinic proposed a model that considers polyps as protruding surfaces and use valley information along with completeness, robustness against spurious responses, continuity, and concavity boundary constraints to generate energy map related to the likelihood of polyp presence Bernal et al. (2015). In Karkanis et al. (2003), the model uses color feature extraction scheme based on wavelet decomposition, producing color wavelet covariance feature. This model then uses Linear Discriminant Analysis with the extracted features to classify image regions in the frames. Other handcrafted feature approaches can be found in Taha et al. (2017).

Deep learning approaches have also been proposed for polyp detection in colonoscopy. In

Park et al. (2015), it uses multi-scale architecture with 3 layers of CNN and 3 layers of max-pooling followed with fully connected layer. Another model uses a slightly different approach in which it uses 3 different extracted features to feed to an ensemble of 3 CNNs Tajbakhsh et al. (2015). The extracted features used are color and texture clues, temporal features, and shape in context. Y-Net Mohammed et al. (2018) combines two encoders following VGG19 network architecture (one is pre-trained on ImageNet and the other is initialized using Xavier normal initializer) which are then followed by decoder layers. This was proposed to solve the problem of small dataset size in medical image analysis.

2.3 Deep Learning Object Detection Categories

Image level classification task performance is affected by the position and size of the object in the frame. Thus, having an object detection stage can improve the classification performance of the model. Early approach of object detection using deep learning was to use sliding window mechanism, where the deep learning classifier is applied to the sliding window to detect object within that window. Then, other approaches for object detection was developed to improve efficiency and accuracy of the models.

General Object Detection models using deep learning can be categorized into two main framework categories; two-stage framework and one-stage framework. The difference between the two categories is mainly due to the region proposal generation stage.

2.3.1 Two-stage Frameworks

In two-stage framework, a region proposal stage is used to generate possible regions of interest. The proposed regions are then passed to a classifier to get the final prediction for each region. Thus, this type of framework has two stages; the first being the region proposal stage, and the second being the classifier stage.

Two-stage framework object detection models generally have higher accuracy compared to one stage framework models. This is because each stage of the framework is optimized to do a specific task. The region proposal stage is trained to optimize the detection and localization of objects with various size ratios and position within the image frame. The classifier is optimized to classify the detected objects. However, two-stage framework models suffer on their processing speed performance. The region proposal stage is often found to be the bottleneck as it is often a slow process.

The development of two-stage framework models begins with R-CNN Girshick et al. (2014). It uses selective search for its region proposal stage. While having good accuracy, R-CNN is slow and far from reaching real-time level performance. Furthermore, training is multistage pipeline (which is slow and difficult) and features from the 2000 region proposals are extracted from CNN separately and required to be stored in the disk. Thus, several improvements and modifications to R-CNN has been proposed to improve its processing speed and accuracy performances. SPPNet He et al. (2015) uses *spatial pyramid pooling layer* to produce fixed size input for the fully connected layer from any size of feature map output. This way, the CNN network processes the entire image once, as opposed to R-CNN which processes each region proposal from the image separately. However, this network still uses a multistage pipeline for training. Fast R-CNN Girshick (2015) uses a similar idea to SPPNet, but it only uses single-level pooling layer (instead of three-level pooling layer like in SPPNet) which they call *RoI pooling layer* (Regions of Interest pooling layer). Furthermore, the model minimizes multi-task loss from softmax layer (for class prediction) and linear regression layer (for bounding box locations). This allows the network to be trained end-to-end, which simplifies the training process. Despite all these improvements, the selective search algorithm for region proposal is still the bottleneck for speed performance for these models. Faster R-CNN Ren et al. (2015) solved this problem by replacing the selective search algorithm with region proposal network to generate region proposals from the image.

2.3.2 One-stage Frameworks

As opposed to two-stage framework, one-stage framework does not have region proposal generation stage, making it single-stage pipeline. This framework formulates object detection as a regression problem and directly predicts bounding box offsets and class probability based on dense sampling of possible locations from the entire image in a single network pass.

One-stage framework has simpler architecture compared to two-stage framework. This framework can also be trained end-to-end, which makes it easy to train. Having simpler architecture, this framework has better speed performance compared to two-stage framework, often reaching real-time performance. However, since this framework uses dense sampling of possible locations of bounding boxes, it often has lower detection accuracy performance compared to two-stage framework.

An example of an early proposed one-stage framework is OverFeat Sermanet et al. (2013), which uses a multi-scale sliding window mechanism to detect objects in the image. Other popular one-stage framework models are YOLO Redmon et al. (2016) and SSD Liu et al. (2016). The very first YOLO model splits the image into $S \times S$ cells. Each cell is then responsible to predict an object existence score, a class probability conditioned on object existence, and B bounding box locations. Here, it only predicts one class for each cell, no matter how many bounding boxes are assigned to each cell. Learned features from backbone CNN layers are passed to fully connected layers to make predictions. The authors of this model have proposed two iterative improvements to YOLO, which they call YOLOv2 Redmon & Farhadi (2017) (sometimes YOLO9000) and YOLOv3 Redmon & Farhadi (2018). YOLOv2 uses some tricks such as batch normalization, convolutional layer for prediction (instead of fully connected layer), class prediction for each bounding box (instead of for each grid cell), anchor box prediction, dimension priors using K-Means, direct location prediction, Darknet-19 as backbone, and other tricks to improve the performance of YOLO. Notable differences between YOLOv3 and YOLOv2 are the use of Darknet-53 as the backbone, multi-scale prediction, and independent logistic classifiers with binary cross entropy loss for class prediction (instead of softmax function with mean squared error loss). Similar to YOLOv3, SSD uses multi-scale prediction and convolutional layer for prediction. Each scale predicts bounding box locations and class predictions for each grid cell in that scale. Other variants of SSD have been proposed to solve known problems in SSD model. RetinaNet Lin et al. (2017) introduces focal loss to handle class imbalance in object detection problem and feature pyramid network to improve the accuracy performance in each scale. RefineDet Zhang et al. (2018) introduces anchor refinement module to produce initial predictions that will be refined by the object detection module. This also filters out negative results that will be passed down to the classifiers.

Chapter 3

General Object Detection Model

In this chapter, we review different general object detection models used in this comparative study. There are six models compared; namely FasterRCNN Ren et al. (2015), YOLOv3 Redmon & Farhadi (2018), SSD Liu et al. (2016), RetinaNet Lin et al. (2017), RefineDet Zhang et al. (2018), and DetNet Li et al. (2018).

3.1 Faster RCNN

Faster RCNN is a two-stage framework model and one of the families of RCNN networks. It improves on the Fast RCNN network by replacing slow selective search algorithm for region proposal generation with region proposal network. This results in faster detection rate. Furthermore, region proposal network is trainable, which can potentially achieve better performance.

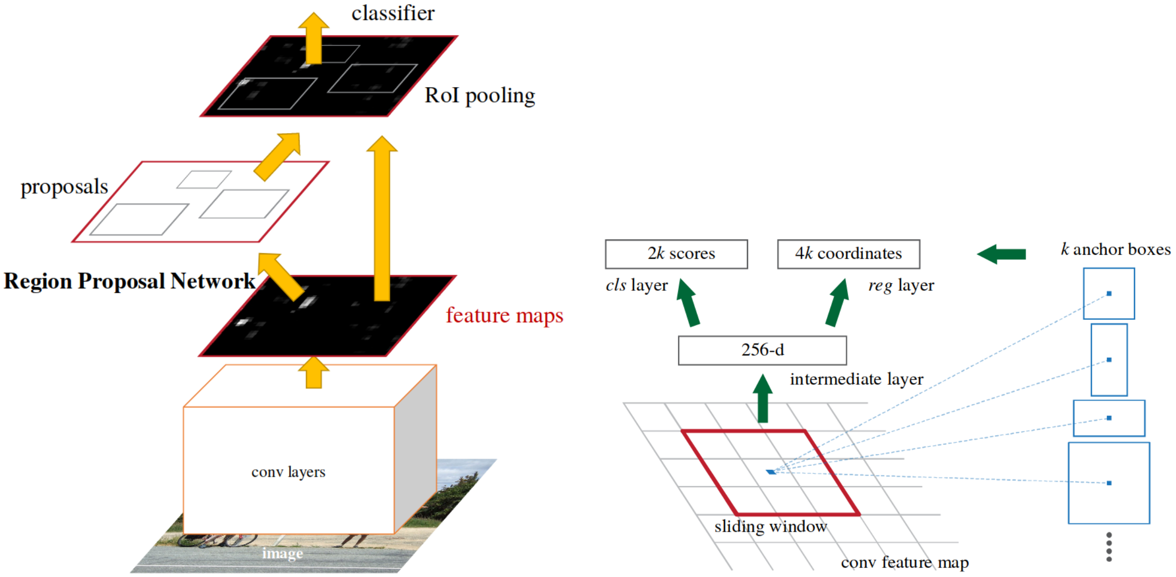
Faster RCNN is composed mainly of two modules, the region proposal network module and the Fast RCNN detector module. Both modules share the same feature maps to simplify computation and make it efficient. First, the backbone network (in this case, it is ResNet 101 He et al. (2016)) extracts features from the image. Then, these features are passed down to the region proposal network. The region proposal network applies $n \times n$ convolutional layer sliding window ($n = 3$ in the paper) followed by 1×1 two sibling convolutional layers to the feature map to regress bounding box locations and 2 probabilities corresponding to object and non-object. Each sliding window predicts k pre-defined anchor boxes (k = 9), centered at the sliding window, with different sizes and ratios to achieve multi-scale learning. The model assigns positive label to (i) anchors with highest IoU overlap with ground-truth box, or (ii) anchors with IoU higher than 0.7

with any ground-truth box. Negative label is assigned to anchors with IoU lower than 0.3 for all ground-truth boxes. Anchor boxes that do not meet these conditions are not included in the training objective. Thus, each sliding window outputs $(4+2) \times k$ values. The loss function for region proposal network is as follows:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*)$$

with p_i as the probability of anchor i being an object (1 for object, 0 for non-object), t_i as the parameterization of bounding box for anchor i as described in Ren et al. (2015), L_{cls} as the log loss over two classes, and $L_{r}eg$ as the smooth L_1 loss in Girshick (2015).

After proposed regions are generated, RoI pooling pool the feature map corresponding to the proposed regions to generate fixed size features to be passed down to the two sibling fully connected layers. The loss function used for the final prediction form this two sibling fully connected layers is similar to the loss function used in region proposal network, noting that the number of class is the same as the the number of object categories in the dataset. Fig. 3.1 depicts the architecture of Faster RCNN.



(a) FasterRCNN Architecture

(b) Region Proposal Network Architecture

Figure 3.1: FasterRCNN model architecture. Source: Ren et al. (2015)

Faster RCNN model has high accuracy, benefiting from optimization from separate stages. However, this model cannot perform in real-time speed, performing at around 5 fps Ren et al. (2015).

3.2 YOLOv3

YOLOv3 (You Only Look Once v3) is the last iterative improvement proposed by the original authors. It improves its performance from previous versions by introducing new backbone network, multi-scale prediction, and modified class prediction loss function.

YOLOv3 uses DarkNet-53 as its backbone network for feature extraction. This backbone network incorporates skip connections to solve the vanishing gradient problem in a deep network and upsampling layer to improve multi-scale prediction by combining (concatenating) features from lower scale with higher scale. This model predicts at three different scales for small, medium, and large objects. Feature map from each scale is passed down to a detection module composed of fully convolutional layers. The detection module splits the image into $S \times S$ grid, depending on the scale. Each cell is responsible to predict ground truth objects with centers located inside the cell. Each cell in the grid predicts $B \times (4+1+C)$ values corresponding to B bounding boxes, 4 values for bounding box locations, 1 value for object confidence, and C values for C classes of objects in the dataset. The bounding boxes are centered at the center of the cell and have predictions of parameterized x, y offsets and h, w. Class prediction is assigned per bounding box instead of per cell in the grid like in YOLOv1. The loss function of the model at scale m (which will be summed for all scale for total loss function) with $S \times S$ cells is as follows:

$$L_{total_m} = \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{i,j}^{obj} L_{bbox}(t_i, t_i^*) +$$

$$\lambda_{obj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{i,j}^{obj} L_{logistic}(p_{obj}, p_{obj}^*) +$$

$$\lambda_{noobj}\sum_{i=0}^{S^2}\sum_{j=0}^{B}1_{i,j}^{noobj}L_{logistic}(p_{obj},p_{obj}^{*})+$$

$$\sum_{i=0}^{S^2}\sum_{j=0}^{B}\sum_{k=0}^{C}L_{logistic}(p_k,p_k^{*})$$

with B as the number of bounding boxes, C as the number of classes, λ_{coord} as the weight for bounding box coordinate loss, $\lambda obj, \lambda_{noobj}$ as the weights for object confidence loss, $L_{logistic}$ as the logistic function with binary cross entropy loss, t_i as 4 values of b_x, b_y, b_w, b_h, and L_{bbox} as sum of squared loss for each parameterized bounding box values as follows:

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h}$$

Here, t_x, t_y are the x, y offsets from cell center, c_x, c_y are the coordinates for cell top left corner, p_w, p_h are prior anchor box width and height. These priors for width and height of the bounding boxes (size and aspect ratio) are pre-computed using K-Means algorithm from the dataset.

YOLOv3 has fast detection rate, achieving real-time performance. However, YOLOv3 often has lower detection accuracy compared to Faster RCNN.

3.3 Single Shot Detector

Single Shot Detector model is a one-stage framework model that uses convolutional layers to predict bounding box locations and class prediction at different feature map scales. This model has simple architecture and can achieve fast detection rate with good accuracy.

The proposed SSD model uses VGG-16 network Simonyan & Zisserman (2014) as the backbone network for feature extraction. Similar to YOLOv3, SSD introduced multi-scale prediction first. SSD also introduced the use of convolutional layer for prediction prior to YOLOv3, since YOLOv1 uses fully connected layers for prediction. The detection layers are fairly simple, only composed of convolutional layers which then predict $B \times (4 + C)$ values for each cell in the grid, where B is the number of anchor boxes and C is the number of classes. Here, the number of classes includes the background class. So, if the highest class prediction for that anchor box is the background class, then the anchor box does not contain any object. The feature maps are progressively passed down to the next layers with downsampling to allow the model to predict various small to large objects from different scales. This eventually results in a larger number of prediction anchor boxes compared to YOLOv1. Each ground truth box is matched to anchor box with the best IoU, and then the anchor boxes are matched with any ground truth box with IoU of more than 0.5. The loss function for SSD is shown as follows:

$$L(x,c,l,g) = \frac{1}{N}(L_{conf}(x,c) + \alpha L_{loc}(x,l,g))$$

with N as the number of matched anchor box, $x_{i,j}^{p}$ as the indicator of matching of anchor box i with ground truth box j of category p, l as the predicted box, g as the ground truth box, and α as the weight for localization loss. $L_{conf}(x,c)$ is softmax loss over multiple classes as follows:

$$L_{conf}(x,c) = -\sum_{i \in Pos}^{N} x_{i,j}^{p} log(\hat{c}_i^p) - \sum_{i \in Neg}^{N} log(\hat{c}_i^0)$$

$$\hat{c}_i^p = \frac{exp(c_i^p)}{\sum_p exp(c_i^p)}$$

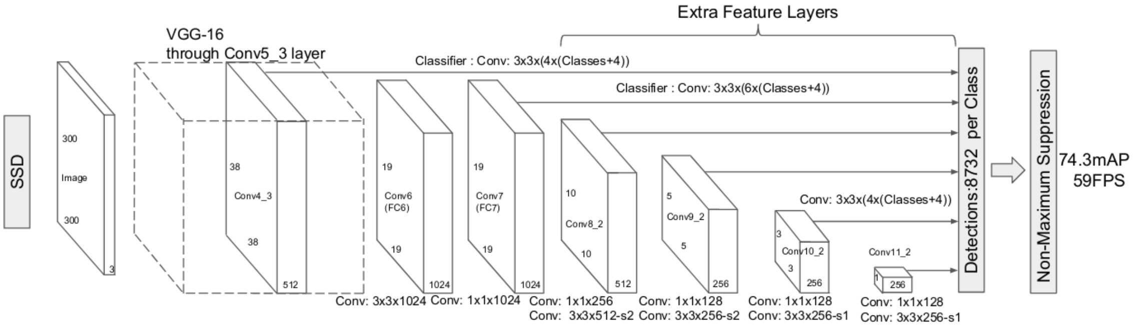
$L_{loc}(x,l,g)$ is a smooth L_1 localization loss for anchor box d as follows:

$$L_{loc}(x,l,g) = \sum_{i=\in Pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{i,j}^{k} smooth_{L_1}(l_i^m - \hat{g}_j^m)$$

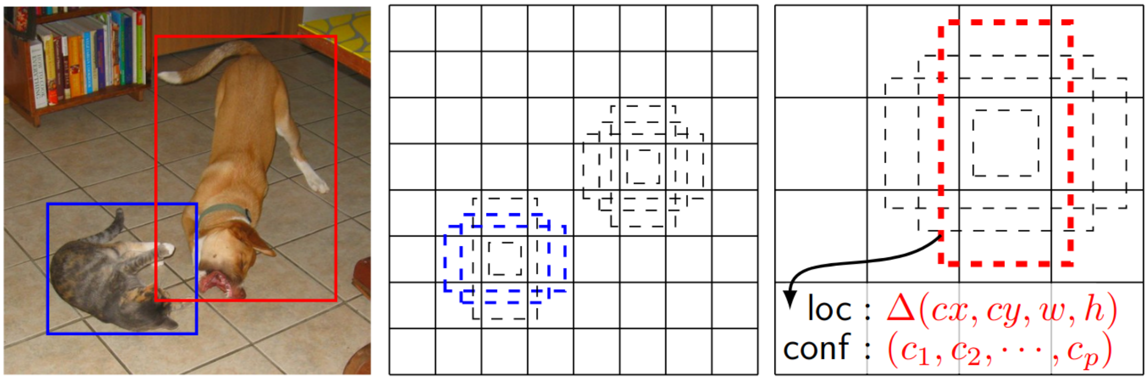
$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^{w} \quad \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^{h}$$

$$\hat{g}_j^{w} = log(\frac{g_j^{w}}{d_i^{w}}) \quad \hat{g}_j^{h} = log(\frac{g_j^{h}}{d_i^{h}})$$

Fig. 3.2 depicts the architecture of SSD.



(a) SSD Architecture



(b) SSD Anchor Box

Figure 3.2: SSD architecture and anchor box. Source: Liu et al. (2016)

SSD has a good trade-off between speed and accuracy. The simple one-stage framework architecture results in fast performance, achieving real-time detection rate. Furthermore, the use of anchor boxes and multi-scale prediction give it good detection accuracy.

3.4 RetinaNet

RetinaNet is a one-stage framework model which is based on SSD model. RetinaNet tries to improve performance by using Feature Pyramid Network for feature extractor and focal loss function to solve the class imbalance problem.

In SSD model, the multi-scale prediction mechanism suffers from its architectural weakness in which higher level layers do not use information from lower level layers. Using feature pyramid network as the backbone, each scale can have better detection accuracy due to the use of both higher level and lower level features. The prediction at each scale of the network is similar to SSD model, predicting for class and bounding box location for each anchor box. Another major difference and contribution of this model is the use of focal loss to solve the class imbalance problem. Class imbalance, specifically extreme background class imbalance, influences one-stage framework detection performance greatly compared to two-stage framework. This is because most background class is implicitly filtered out by the region proposal network as opposed to one-stage framework. The proposed focal loss is as follows. Let us begin by defining the cross-entropy loss for binary classification:

$$\mathrm{CE}(p,y) = -log(p_t), \text{where } p_t = \begin{cases} p & \text{if } y = 1 \\ 1-p & \text{otherwise} \end{cases}$$

The focal loss is then defined as follows:

$$\mathrm{FL}(p_t) = -\alpha_t(1-p_t)^{\gamma} log(p_t)$$

The α_t parameter is the weight for the class to balance out positive/negative examples. γ is the focusing parameter that smoothly adjusts the rate to down-weight easy examples. This way, well classified examples will have small loss value contribution, while missclassified or hard classified examples will have large loss value contribution. Fig. 3.3 depicts the architecture of RetinaNet.

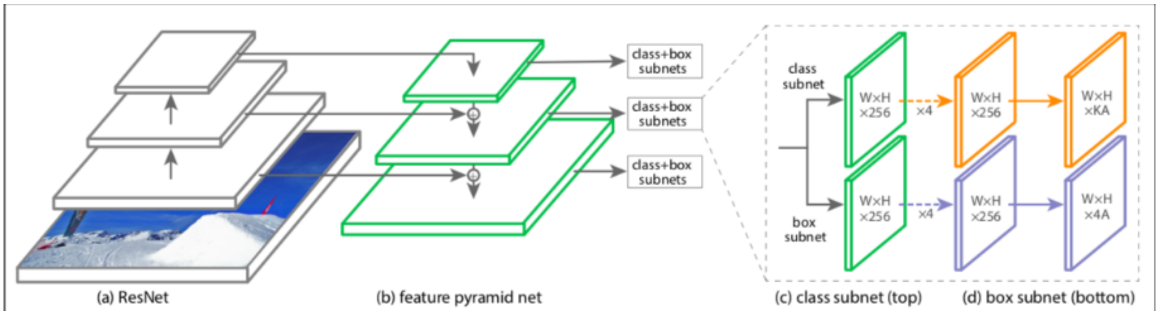


Figure 3.3: RetinaNet architecture. Source: Lin et al. (2017)

Having similar architecture to SSD model for its detection layers, RetinaNet has similar detection speed to SSD. It has better accuracy thanks to the use of focal loss and feature pyramid network (which combines higher level and lower level information).

3.5 RefineDet

RefineDet is a one-stage framework model that is also based on SSD model. Like RetinaNet, this model also aims to solve the class imbalance problem that affects one-stage framework models. However, it achieves it through a different approach than RetinaNet. Inspired by the architectural advantage of two-stage framework models to handle class imbalance, RefineDet combines the architectural advantages of one-stage framework with two-stage framework by using two interconnected modules, namely Anchor Refinement Module (ARM) and (Object Detection Module).

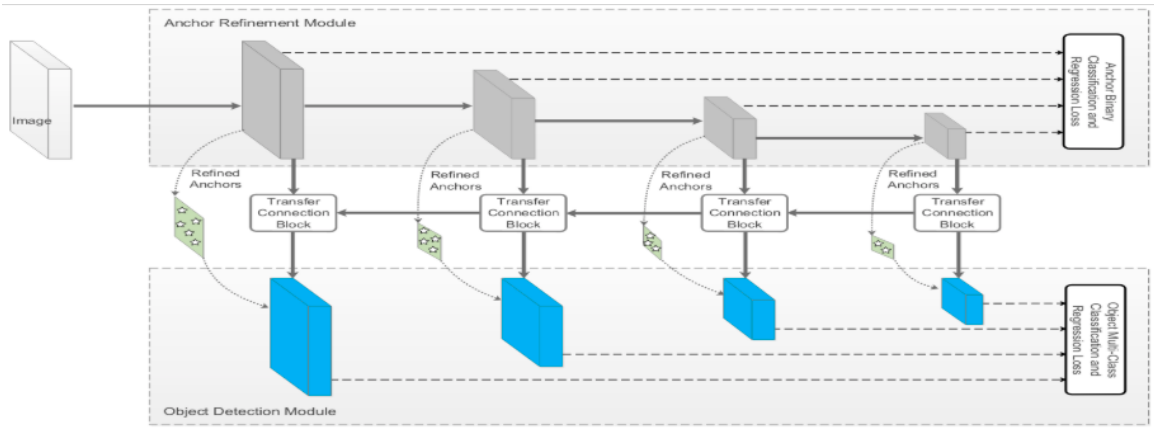
RefineDet uses either VGG-16 network or ResNet-101 network as its backbone for feature extraction. The Anchor Refinement Module is used to refine the initial anchor boxes locations to provide better initialization of coarse bounding boxes locations for final prediction. It also filters out easy negative anchors using binary classification prediction, which in turn reduces search space for the final classifier, similar to two-stage framework models. Similar to SSD model, each scale divides the image into grid cells. Each cell predicts $B \times (4+2)$ values corresponding to B bounding boxes, 4 values corresponding to bounding box locations, and 2 values corresponding to confidence scores for the presence of foreground object in that anchor box. After obtaining refined anchor boxes, the corresponding feature map for each scale and the refined anchor boxes containing foreground objects are passed down to the Object Detection Module through Transfer Connection Block module. Transfer Connection Block modules convert features from different layers from Anchor Refinement Module to the correct input dimension for the corresponding feature map scale. It uses deconvolution operation on features from higher level layer and sums them element-wise with the corresponding feature map scale. This way, the model combines higher level and lower level contextual information to make better predictions. Finally, the Object Detection Module makes the final predictions of objects in the image and their locations. Again, this module has similar architecture to SSD detection layer, which splits the image into grid cells for each scale. For each cell, it predicts $B \times (4+C)$ values where C is the number of classes. The loss function for the model is as follows:

$$L(p_i, x_i, c_i, t_i) =$$

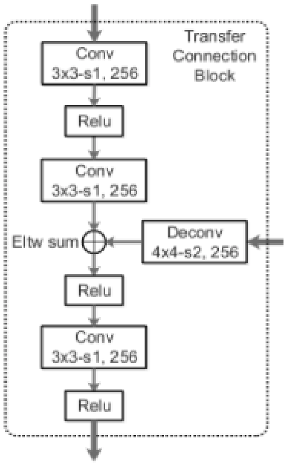
$$\frac{1}{N_{arm}}(\sum_i L_b(p_i, [l_i^* \geq 1]) + \sum_i [l_i^* \geq 1] L_r(x_i, g_i^*)) +$$

$$\frac{1}{N_{odm}}(\sum_i L_m(c_i, l_i^*) + \sum_i [l_i^* \geq 1] L_r(t_i, g_i^*))$$

L_b is the binary classification loss (cross-entropy loss of two classes; object vs non-object), L_m is the softmax multiclass cross-entropy loss, L_r is the smooth L_1 regression loss of the bounding box as in FasterRCNN Ren et al. (2015), i refers to the anchor box i, l_i* is the ground-truth class label, p_i is the prediction of objectness confidence in ARM, c_i is the class prediction in ODM, x_i is the refined prediction of bounding box location and size in ARM, t_i is the final prediction of bounding box location and size in ODM, g_i* is the ground-truth bounding box location and size, N_{arm} is the number of positive anchor boxes in ARM, N_{ODM} is the number of positive anchor boxes in ODM, and $[l_i \geq 1]$ is 1 when anchor is not negative and 0 otherwise. Fig. 3.4 depicts the architecture of RefineDet.


Anchor Refinement Module
Image
Anchor Binary Classification and Regression Loss
Refined Anchors
Transfer Connection Block
Object Multi-Class Classification and Regression Loss
Object Detection Module

(a) RefineDet Architecture



(b) RefineDet Transfer Connection Block Module

Figure 3.4: RefineDet architecture and Transfer Connection Block Module. Source: Zhang et al. (2018)

RefineDet has good speed performance similar to SSD model, achieving real-time performance. It also has better accuracy thanks to its architectural design that removes easy negative examples prior to final classification. The refined anchor boxes also provide better initialization for final bounding box location prediction.

3.6 DetNet

DetNet is different than previously discussed general object detection models. DetNet is not an object detection model, rather it is a backbone network specifically designed for object detection problem. It was motivated by the popular use of pre-trained backbone network on ImageNet classification problem for object detection. The authors argued that previously proposed backbone networks that are pre-trained on image level classification task are not well suited for object detection task due to the increasing loss of spatial information in higher level layers from down-sampling operations. Previously proposed backbone networks often have different numbers of stages resulting in different spatial resolutions, weak visibility of large objects due to large strides (which reduces object localization ability), and invisibility of small objects as larger contextual information is integrated.

The proposed DetNet 59 backbone network follows the same settings as ResNet-50 for the first 4 stages. In stage 5 and 6, the spatial resolutions are fixed by using dilated bottleneck with 1×1 convolutions at the beginning of each stage. This allows the network to enlarge its receptive field while keeping the spatial resolution the same as in stage 4. Due to the expensive operation of dilated convolution, the numbers of channels in stage 5 and 6 are kept the same as in stage 4. The network can also use FPN-like design by summing up features from higher level layers with lower level layers. Fig. 3.5 depicts the bottleneck blocks used in DetNet. The structure of DetNet is depicted in 3.6.

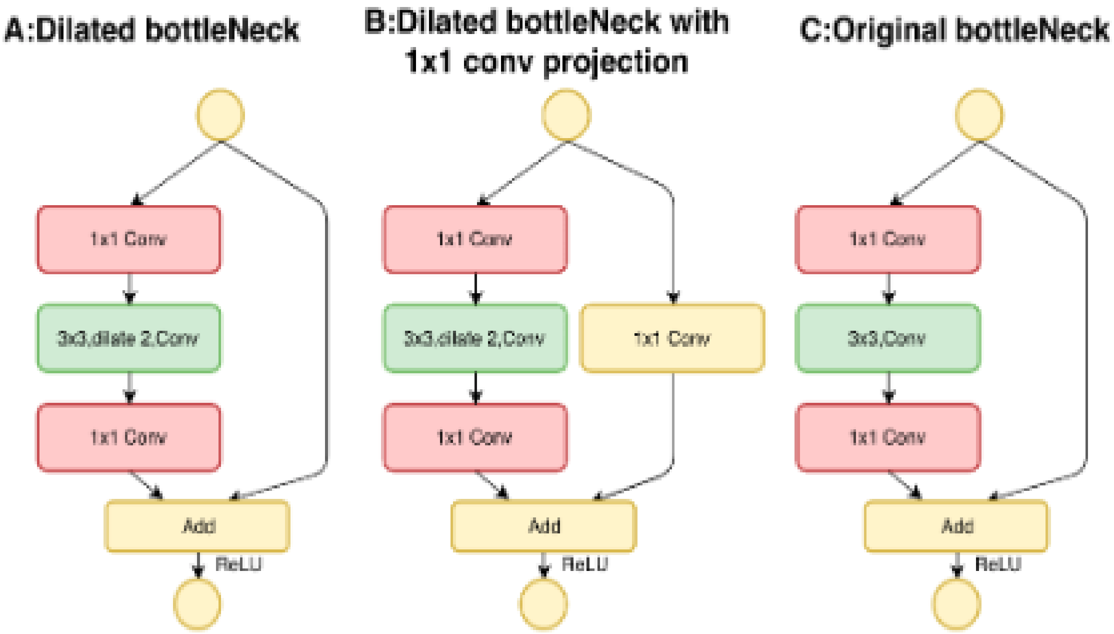


Figure 3.5: DetNet architecture. Source: Li et al. (2018)

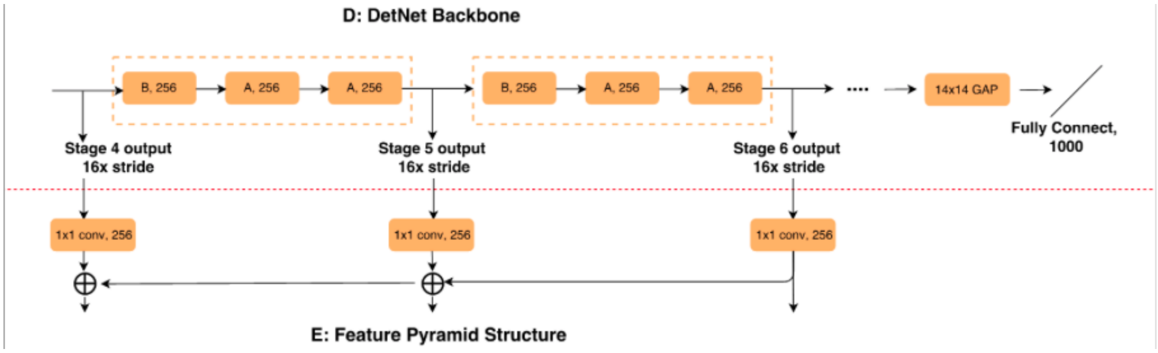


Figure 3.6: DetNet architecture. Source: Li et al. (2018)

DetNet as backbone network improves the performance of previously proposed generic object detection model.

Chapter 4

Experimentation

In this chapter, we explain the experiment settings as well as the dataset that we use for experiments in this comparative study.

4.1 Dataset

For this comparative study, we created a new dataset by combining three existing polyp detection datasets and a new dataset containing 80 unannotated video sequences. Brief overview of each dataset is explained as the following.

4.1.1 MICCAI 2017: GIANA Endoscopic Vision Challenge Dataset

This dataset is a part of MICCAI 2017 Endoscopic Vision Challenge Gastrointestinal Image ANAlysis (GIANA) Sub-Challenge for Polyp Detection task. The original task for this dataset is to detect and locate the presence/absence of polyps in a frame from colonoscopy video. It is composed of 18 short videos for training and more than 20 short and long videos for testing. Each frame in the training set has associated ground truth segmentation mask for the polyps within it.

4.1.2 CVC-ColonDB Dataset

This dataset comes from Bernal et al. (2012). It contains 15 short colonoscopy video sequences from 15 different studies. The original task for this dataset is to detect and locate the presence/absence of polyps in a frame from colonoscopy video. There is a total of 300 frames in this dataset. Each frame in this dataset has associated ground truth segmentation mask for the polyps within it.

4.1.3 Gastrointestinal Lesions in Regular Colonoscopy Dataset

This dataset comes from Mesejo et al. (2016). It contains a total of 76 video sequences; 15 serrated adenomas, 21 hyperplastic lesions, and 40 adenoma video sequences. Each video sequence was recorded using Narrow-Band Imaging and White Light colonoscopy. However, this dataset does not come with ground truth segmentation mask nor ground truth polyp bounding box mask. The original task for this dataset is to classify each video sequence to its correct polyp category. Particularly, the original authors were interested to maximize accuracy while minimizing false positives and false negatives.

4.1.4 KUMC 80 Videos Dataset

This dataset comes from the University of Kansas Medical Center. It contains 80 colonoscopy video sequences. Each video sequence has been inspected and labeled by an endoscopist for two polyp categories; hyperplastic and adenomatous. However, this dataset does not come with ground truth segmentation mask nor ground truth polyp bounding box mask.

4.2 Dataset Preparation

Since we use a combination of previously mentioned 4 colonoscopy datasets, we need to prepare the datasets for our experiment. The followings are the data preparation steps that we took.

4.2.1 Ground Truth Bounding Box Annotation

As previously mentioned, KUMC 80 Videos Dataset and Gastrointestinal Lesions in Regular Colonoscopy Dataset do not have ground truth annotation for polyp location in the frames. Thus, we manually annotated bounding boxes for the polyps that we found in the frames. The annotations are stored in PASCAL VOC format Everingham et al. (2010b).

The MICCAI 2017 GIANA Endoscopic Vision Challenge Dataset and CVC-ColonDB Dataset, on the other hand, contain ground truth segmentation mask for polyps in the frames. We converted the segmentation mask to bounding box by recording the maximum and minimum in X and Y coordinate of the frames. Finally, we store bounding boxes in PASCAL VOC format.

4.2.2 Frame Selection

In order to make a fair comparison for the performances of the models in each video sequence, we tried to balance the number of frames in each video sequence. Each video sequence contains different numbers of frames, depending on the length of the video, and can vary from 300 - 1500 frames. The exception being the CVC-ColonDB which only contains a total of 300 frames for the 15 video sequences. Due to this variation, we tried to balance the number of frames for the three other datasets by filtering the frames.

Firstly, we filtered the frames by skipping some frames in the sequence so that we get a reasonably balanced number of frames per video sequence. This is also because most of the video sequences were recorded by trying to stay still and focused on the polyps, thus, there is very little variation within subsequent frames. Skipping some frames in the sequence makes sure that we have high variations of the polyp appearances from different angles.

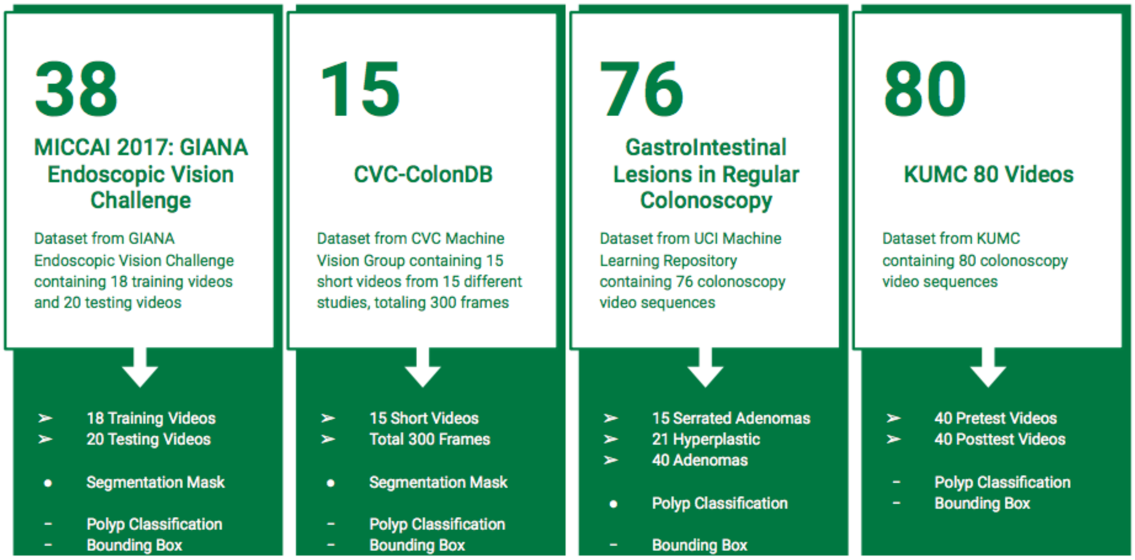
Secondly, we removed blurry and bad frames from the sequences. During the colonoscopy procedure, it is not uncommon for gastroentrologists to move the colonoscope view around to get different angles of the polyp. Sometimes the movement is too quick that makes the resulting recorded frames to be blurry. Furthermore, sometimes gastroentrologists try to get as much detail as possible about the polyp appearance by zooming and closing in to the polyp. This causes the view of the polyp to be too close and cover the whole frame. Thus, we removed blurry and bad frames to get a cleaner dataset.

Thirdly, we removed multiple polyp frames in the sequence. The MICCAI 2017 GIANA Endoscopic Vision Challenge Dataset and CVC-ColonDB Dataset do not have ground truth for the class category of the polyps. Thus, we asked for an endoscopist to categorize the video sequences into hyperplastic and adenomatous. However, sometimes the video sequences zoom out of the dominant polyp in the sequence to either try to get different angles of the same polyp or to find other possible polyps in the colon. Furthermore, some frames also contain multiple polyps. To be sure that the polyp in the frame is of its correct class category, we removed frames that have multiple polyps in it. We also removed frames that contain different polyps other than the dominantly focused polyp in the video sequence.

Lastly, we removed outlier video sequences. Some of the video sequences contain multiple polyps for the entire frames, which we decided to remove from the dataset. Also, some of the video sequences contain polyps that have very distinct appearances than the rest of the polyps and we consider these sequences to be too difficult to classify due to their unique appearances. So, we decided to remove such sequences.

4.2.3 Dataset Split

To measure and compare the performances of each model, we first split the sequences per dataset into 70/30 split for training and test. So, an example is for KUMC 80 Videos dataset, we split the sequences into 70/30 split. We split the datasets per sequence because we want to see how the models perform on unseen colonoscopy video sequences, which potentially contain polyps of various appearances. After we get 70/30 splits per video sequences for each dataset for training and test datasets, we further split the training datasets into training and validation datasets with 70/30 split. Lastly, we group the dataset splits into training, validation, and test datasets.



38
MICCAI 2017: GIANA Endoscopic Vision Challenge
Dataset from GIANA Endoscopic Vision Challenge containing 18 training videos and 20 testing videos
18 Training Videos
20 Testing Videos
Segmentation Mask
Polyp Classification
Bounding Box
15
CVC-ColonDB
Dataset from CVC Machine Vision Group containing 15 short videos from 15 different studies, totaling 300 frames
15 Short Videos
Total 300 Frames
Segmentation Mask
Polyp Classification
Bounding Box
76
GastroIntestinal Lesions in Regular Colonoscopy
Dataset from UCI Machine Learning Repository containing 76 colonoscopy video sequences
15 Serrated Adenomas
21 Hyperplastic
40 Adenomas
Polyp Classification
Bounding Box
80
KUMC 80 Videos
Dataset from KUMC containing 80 colonoscopy video sequences
40 Pretest Videos
40 Posttest Videos
Polyp Classification
Bounding Box

Figure 4.1: Summary of datasets

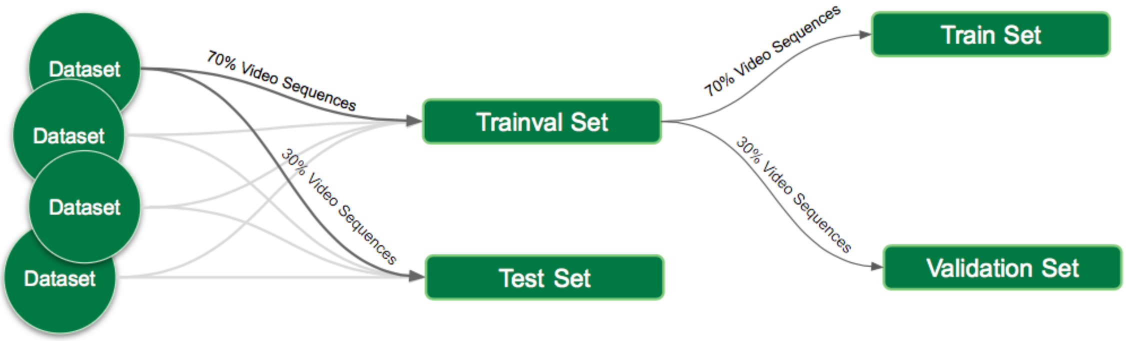


Figure 4.2: Visualization of dataset split

4.2.4 Dataset Summary

Figure 4.1 depicts summary about the datasets used in this study. Figure 4.2 visualizes the steps to split and combine the existing datasets to generate our own dataset for this study.

4.3 Experiment Settings

The experiment was conducted by training the 6 models discussed previously in General Object Detection Model chapter using the training dataset. The validation dataset was used to monitor the performance of each model during training to avoid overfitting. We used the mean average precision metric for validation. Finally, the test dataset was used to compare the performances of the trained models. The datasets were resized to fit each model's requirement for image size. Furthermore, data augmentation techniques such as patch cropping as well as random horizontal and vertical flipping were performed.

4.3.1 Model Hyperparameter Settings

The followings are the hyperparameters we used for training and evaluating the models:

- **DetNet**: This model was trained using learning rate of 1×10^{-3}, weight decay rate of 1×10^{-4}, batch size of 128, image size of 600×600, and total epoch of 7. Evaluation was done using NMS threshold of 0.45 and confidence threshold of 0.5
- **FasterRCNN**: This model was trained using learning rate of 1×10^{-3}, weight decay rate of 1×10^{-1}, batch size of 8, image size of 600×600, and total epoch of 30. Evaluation was done using NMS threshold of 0.45 and confidence threshold of 0.5
- **RefineDet**: This model was trained using learning rate of 1×10^{-4}, weight decay rate of 5×10^{-4}, batch size of 8, image size of 512×512, and total iteration of 175000. Evaluation was done using NMS threshold of 0.45 and confidence threshold of 0.5
- **RetinaNet**: This model was trained using learning rate of 1×10^{-5}, weight decay rate of 0, batch size of 1, image size of 608, and total epoch of 100. Evaluation was done using NMS threshold of 0.5 and confidence threshold of 0.05
- **SSD**: This model was trained using learning rate of 4×10^{-4}, weight decay rate of 1×10^{-4}, batch size of 32, image size of 300×300, and total epoch of 50. Evaluation was done using NMS threshold of 0.45 and confidence threshold of 0.5

- **YOLOv3**: This model was trained using learning rate of 1×10^{-3}, weight decay rate of 5×10^{-4}, batch size of 32, image size of 416×416, and total iteration of 44000. Evaluation was done using NMS threshold of 0.45 and confidence threshold of 0.3

4.4 Comparison Methods

We use the following methods and metrics to compare the performances of the models. Particularly, we are interested to measure the performances regarding polyp detection, polyp localization, and polyp classification. The followings are further explanation of each method.

4.4.1 Polyp Detection

Here, we compare each model's ability to correctly classify polyp category within an image frame. In this comparison method, we do not take the bounding boxes generated by the models into consideration. Thus, we regard this computer vision problem as an image classification task. Specifically, we define the following criteria:

- **True Positive:** model correctly predicts particular polyp category for the image
- **True Negative:** model correctly predicts the image does not contain particular polyp category (or no polyp)
- **False Positive:** model predicts particular polyp category while image does not contain the predicted polyp category (or no polyp)
- **False Negative:** model predicts wrong polyp category (or no polyp) while image contains particular polyp category

4.4.2 Polyp Localization

Here, we compare each model's ability to correctly classify polyp category as well as to correctly predict the size and location of the polyp within an image frame. Thus, we regard this computer vision problem as an object detection task. We use Intersection Over Union (IoU) value between the ground truth bounding box and the prediction bounding box to determine whether a prediction bounding box correctly predicts the size and location of the polyp. Specifically, we define the following criteria:

- **True Positive:** IoU of ground truth bounding box and prediction bounding box > IoU threshold as well as correct polyp category prediction
- **True Negative:** Correct no prediction bounding box for no ground truth bounding box. In object detection task, the there are infinitely many possible True Negative, thus it is rarely used
- **False Positive:** Prediction bounding box has IoU < IoU threshold with all ground truth bounding box (or no ground truth bounding box), wrong polyp category prediction with matching ground truth bounding box
- **False Negative:** No prediction bounding box has IoU > IoU threshold with ground truth bounding box

4.4.3 1 Class Polyp Classification VS 2-Classes Polyp Classification

Here, we consider comparing the models under two scenarios, 1-class polyp classification or 2-classes polyp classification. In 1-class polyp classification, we regard the predictions as polyp or non-polyp. In 2-classes polyp classification, we further consider whether the polyp is adenomatous or hyperplastic. We use 1-class polyp classification to see how the models perform if we disregard the polyp categories. We apply these two scenarios for both polyp detection and polyp localization methods.

4.4.4 Still Frame vs Video Sequence

Here, we consider the performances of the models to correctly predict still frames and the whole video sequence. Under the still frame scenario, we consider each image as an individual image with no relation to all the other image frames in the test dataset. Thus, it measures the performances of the models, given random colonoscopy image frame. Under the video sequence scenario, we classify the video sequence based on the mostly predicted polyp category; adenomatous or hyperplastic. So, we have the final polyp category as $\arg\max x := \{\frac{N_x}{N_p} | x \in S_c\}$ where N_x is the number of frames predicted as x polyp category, N_p is the total number of frames containing polyp prediction in the video sequence, and S_c is the set of polyp category. Furthermore, these two comparison scenarios only apply to 2-classes polyp classification.

4.4.5 Combined Performance

Here, we try to see the combined performance of all the models, considering only the true positive cases. Specifically, we measure the recall value using the combined true positive cases from all the models. We are interested to see the optimistic performance of current object detection models if we combine them together. This shows possible performance improvements if we take the best predictions of each model. We define the combined performance as all the ground truth bounding boxes that have been matched (correct IoU and classification) with predictions from all the models combined. So, essentially, this is just all the bounding boxes that have been predicted successfully by any of the model, and we treat them as predictions from a probable model.

4.4.6 Comparison Metrics

In this subsection, we define metrics we use to measure the performances of the models.

- **Intersection over Union (IoU) or Jaccard Index:** Area of Intersection over Union between ground truth bounding box and prediction bounding box

$$\mathrm{IoU}(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

- **Precision:** Fraction of true positive over all positive predictions

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}}$$

- **Recall:** Fraction of true positive over all positive cases in the set

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

- **Average Precision:** Average of precision values given corresponding recall values over recall = 0 to recall = 1. This is essentially the area under precision-recall curve.

$$\text{Average Precision} = \sum_{k=1}^{n} \text{Precision}(k)\ \Delta\text{Recall(k)}$$

To minimize the impact of the change of precision values over a large number of different recall values, we can also use interpolated average precision over some fixed values *m* by taking the maximum of the precision values greater than *i*. So, we have

$$\text{Interpolated Average Precision} = \frac{1}{m} \sum_{i \in \{\frac{x}{m} | x=1,2,\ldots,m\}} \max_{\tilde{i}:\tilde{i} \geq i} \text{Precision}(i)$$

- **F1:** Harmonic mean of precision and recall values

$$\text{F1} = \frac{\text{2(Precision) (Recall)}}{\text{Precision} + \text{Recall}}$$

- **False Positive Rate:** Fraction of false positive over all negative cases in the set

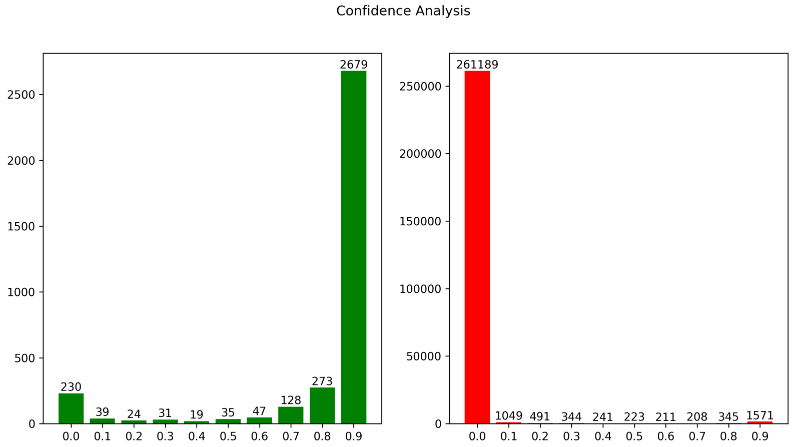
$$\text{False Positive Rate} = \frac{\text{False Positive}}{\text{False Positive} + \text{True Negative}}$$

- **Precision-Recall Curve:** Curve of precision values given recall values
- **ROC Curve:** Curve of true positive rate values (recall values) given false positive rate values

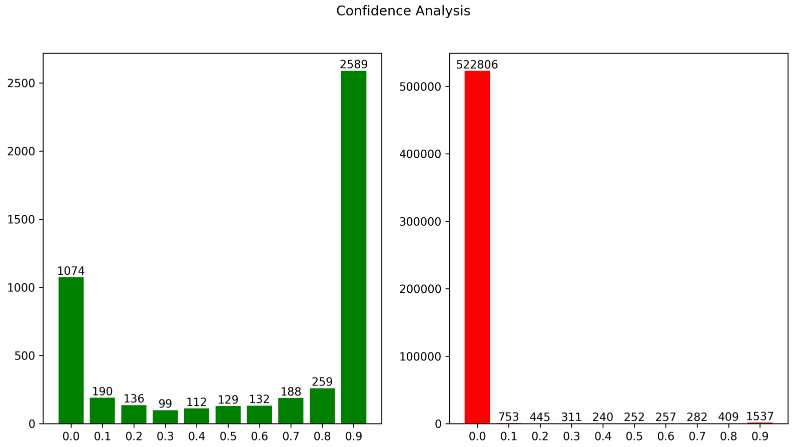
Chapter 5

Results and Discussions

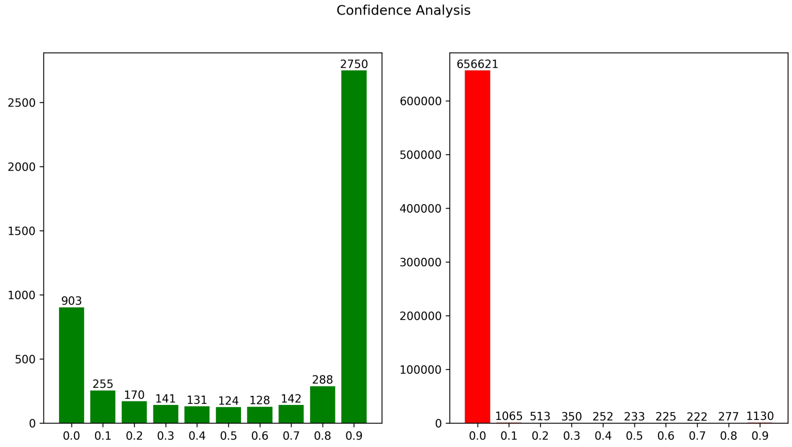
5.1 Confidence Score Analysis



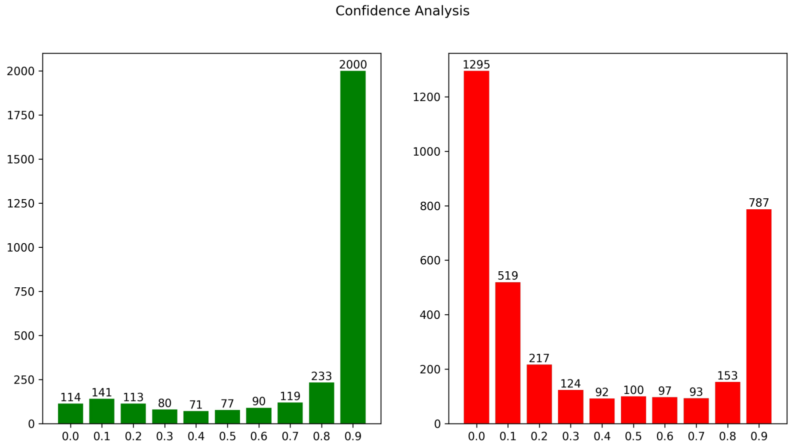
(a) Confidence Analysis: DetNet



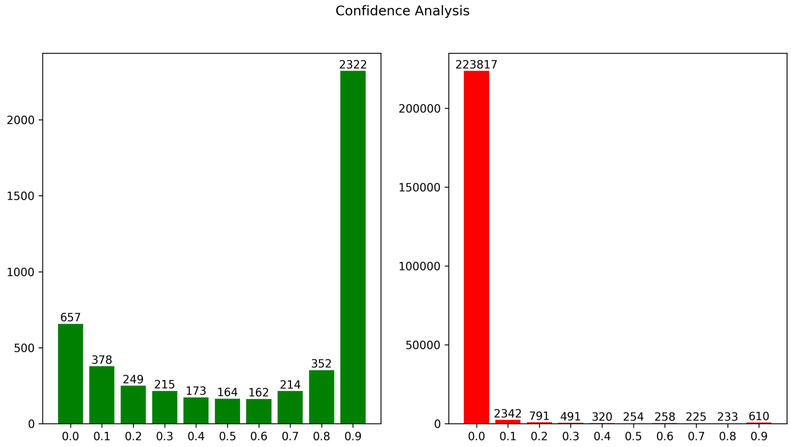
(b) Confidence Analysis: FasterRCNN



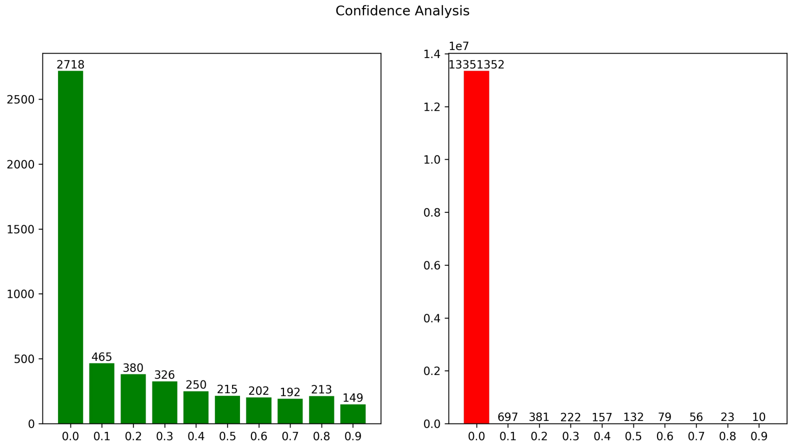
(c) Confidence Analysis: RefineDet



(d) Confidence Analysis: RetinaNet



(e) Confidence Analysis: SSD



(f) Confidence Analysis: YOLOv3

Figure 5.0: Distribution of prediction confidence scores for each model, green shows correct prediction, while red shows false positive prediction

Before we analyze the prediction results of each model further, let us analyze the confidence scores for predictions made by the models. The confidence threshold we set in the post-processing step will affect the predictions to be kept for final predictions. Figures in 5.0 show the distributions of the predictions' confidence scores made by each model. The green bar plots on the left show the distribution of correct class, size, and location predictions of the ground truth bounding boxes. The red bar plots on the right show the distribution of false positive predictions (wrong class, size, or location predictions).

As we can see, 5 of the 6 models, excluding YOLOv3, have the highest number of correct predictions with confidence score between 0.9 – 1.0. Furthermore, 4 of them have "J" shape distribution, with the second highest number of correct predictions having confidence score between 0.0 – 0.1. The false positive distributions with confidence score between 0.9 - 1.0 also have small number of false positive. This means, that when the models make predictions with high scores, those predictions are quite likely to be correct. Having lower confidence score threshold allows us to predict more correct bounding boxes. However, as we can see from the corresponding false positive distributions, it introduces more false positive predictions (with the highest number having confidence score between 0.0 - 0.1). This surely is not a result that we want since it will not be usable in the real world.

Having a lower confidence threshold can also be misleading when we look at the mean average precision score. Due to the way we implement mean average precision computation, lower confidence score threshold will mostly result in a higher score even though it introduces more false positive (which should affect precision). Since we usually sort the predictions by confidence score before computing mean average precision, then lower confidence predictions will always be considered last. This results in high precision values at low recall values, which will results in high mean average precision even though it has many false positive predictions with low confidence scores. Furthermore, from the average precision formula, it always increases. So, as long as it hits correct predictions at low confidence scores, the mAP will still increase. Thus, we need to understand how the confidence score threshold affects mean average precision.

The YOLOv3 confidence score distribution is different from the other distributions since it keeps decreasing with an increasing confidence score. This means that if we set a higher confidence score threshold, the model will have fewer correct predictions, which in turn results in a lower mean average precision score.

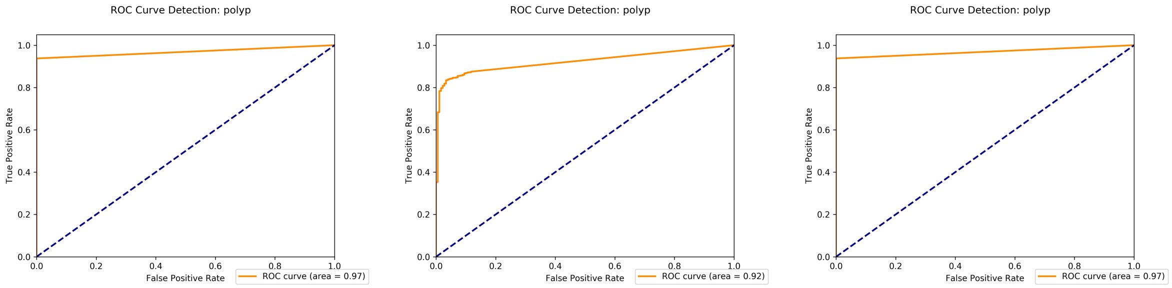
RetinaNet has the smallest number of false positive predictions compared to the other models. This is probably because RetinaNet benefits from the use of Focal Loss, which helps with easy background examples. Thus, it is less likely to make false positive predictions.

RefineDet has the highest number of correct predictions with confidence score ≥ 0.5, followed by FasterRCNN, and then SSD. This means that these models are able to make correct predictions with high confidence scores.

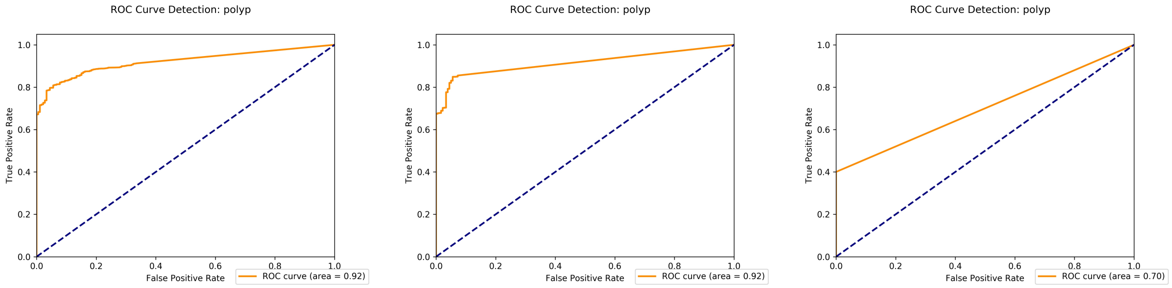
5.2 1-class Polyp Image Classification Task

Metrics	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
Acc	87.0	93.9	87.5	93.9	90.4	85.7	42.0
Bal Acc	93.3	96.8	87.8	96.8	78.8	89.1	70.0
Prec	100	100	99.5	100	98.7	99.7	100
Rec	86.6	93.7	87.4	93.7	91.2	85.4	40.0
F1	92.8	96.7	93.1	96.7	94.8	92.0	57.1
AP	99.5	99.7	99.5	99.7	99.6	99.4	97.9
AUC	93.3	96.8	92.2	96.8	92.0	91.5	70.0

Table 5.1: Performance of each model under different metrics for 1-class polyp detection task (classifying each frame as polyp or non-polyp)

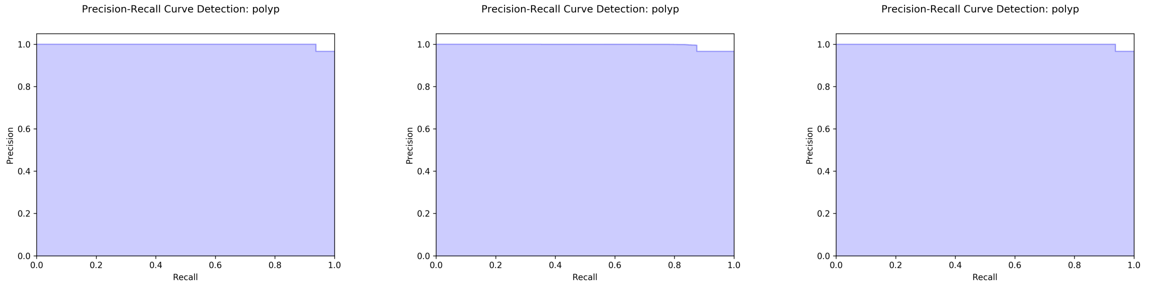


(g) DetNet (h) FasterRCNN (i) RefineDet

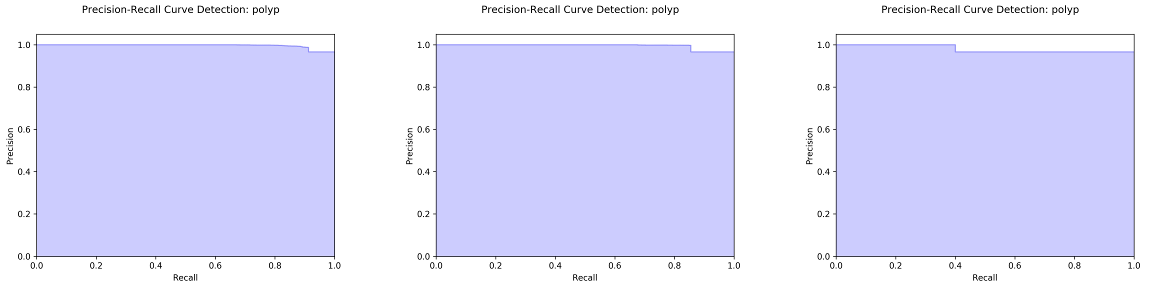


(j) RetinaNet (k) SSD (l) YOLOv3

Figure 5.1: ROC Curve for 1-class polyp detection



(a) DetNet (b) FasterRCNN (c) RefineDet



(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.2: PR Curve for 1-class polyp detection

Here, we consider the task as a 1-class image classification task. So, we do not consider the size and position predictions made by the models. We are only interested in whether the models know that there is a polyp or not in the image frame. Table 5.1 shows different metric scores for the models. Figures in 5.1 show the ROC curve for each model. Figures in 5.2 show the Precision-Recall curve for each model.

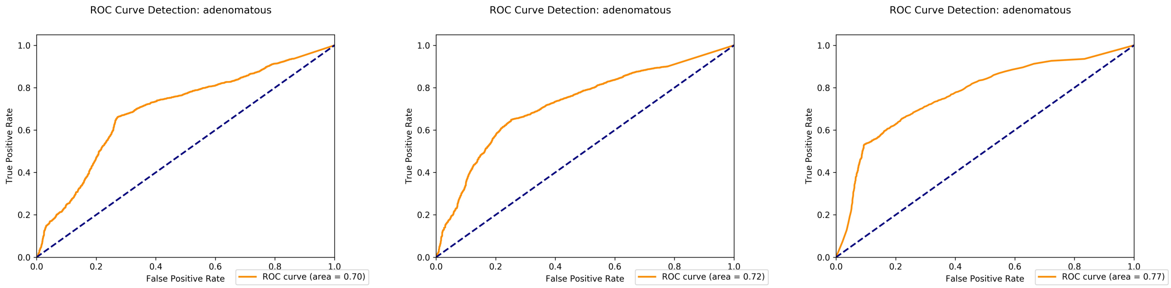
We can see that RefineDet, DetNet, and YOLOv3 have 100 % precision score. This means that these models never make prediction that the image contains polyp when in fact it does not. However, YOLOv3 makes a fewer number of predictions compared to the other models, so this is not particularly interesting. RefineDet and DetNet also have high recall scores (which results in high F1 scores) compared to the other models. So is the case with the average precision and area under ROC curve scores for these two models. So, RefineDet and DetNet are particularly good at knowing if the image frame has polyp or not.

Another interesting thing to note here is that RefineDet and DetNet have better performances than the combined performance. This is because the combined performance only considers correctly predicted bounding boxes. If the models predict that there is a polyp in the image but the prediction is wrong (incorrect class, size, or location), then the prediction will not be included in the combined performance. Thus, combined performance has lower recall score compared to RefineDet and DetNet at 1-class polyp image classification task.

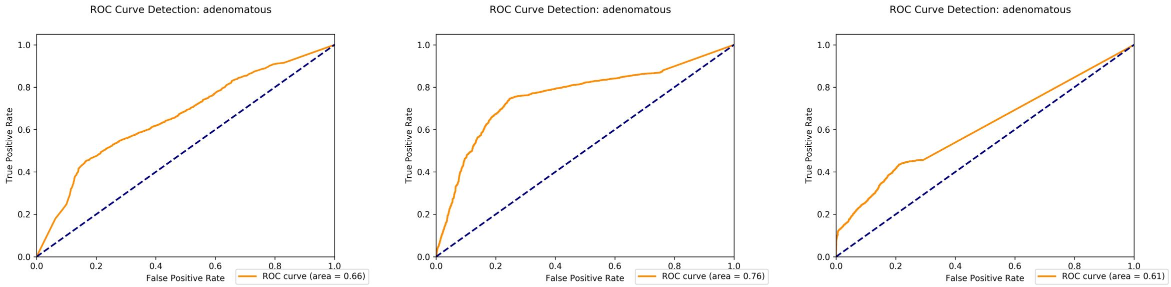
5.3 2-classes Polyp Image Classification Task

Metrics	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
Acc	87.0	66.5	62.4	66.4	63.9	67.8	31.7
Bal Acc	91.3	76.7	69.8	78.8	64.4	74.2	52.0
Prec Ad	100	76.9	77.9	88.4	76.2	80.5	84.0
Prec Hy	100	60.9	59.5	57.9	60.1	73.4	56.5
mPrec	100	68.9	68.7	73.2	68.2	77.0	70.2
Rec Ad	84.9	66.1	64.8	53.1	65.5	74.5	34.2
Rec Hy	89.1	64.2	56.5	83.5	61.4	55.5	21.9
mRec	87.0	65.1	60.6	68.3	63.4	65.0	28.1
F1 Ad	91.8	71.1	70.8	66.4	70.4	77.4	48.6
F1 Hy	94.2	62.5	57.9	68.4	60.7	63.2	31.6
mF1	93.0	66.8	64.4	67.4	65.6	70.3	40.1
AP Ad	93.2	73.7	76.9	79.9	71.8	80.5	69.8
AP Hy	92.4	56.4	57.2	69.0	54.4	62.4	44.3
mAP	92.8	65.1	67.1	74.5	63.1	71.4	57.1
AUC Ad	85.7	69.5	72.3	76.8	66.2	75.9	60.8
AUC Hy	89.6	69.0	65.4	76.4	64.8	69.7	47.0
mAUC	87.7	69.2	68.9	76.6	65.5	72.8	53.9

Table 5.2: Performance of each model under different metrics for 2-classes polyp detection task (classifying each frame as adenomatous, hyperplastic, or non-polyp)

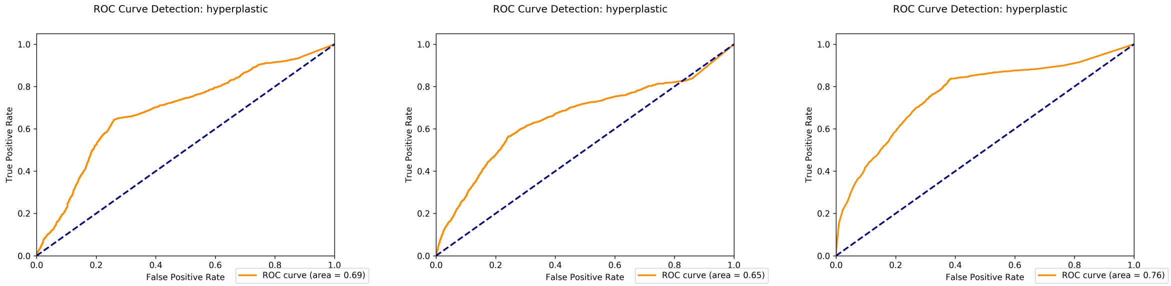


(a) DetNet (b) FasterRCNN (c) RefineDet

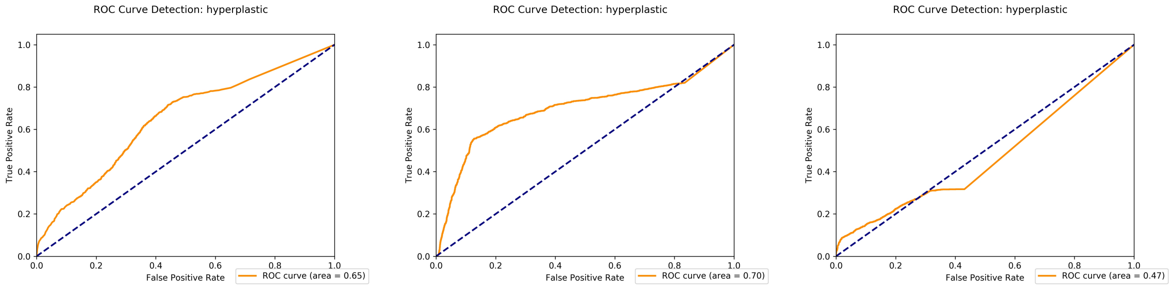


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.3: ROC Curve for 2-classes polyp detection: adenomatous polyp

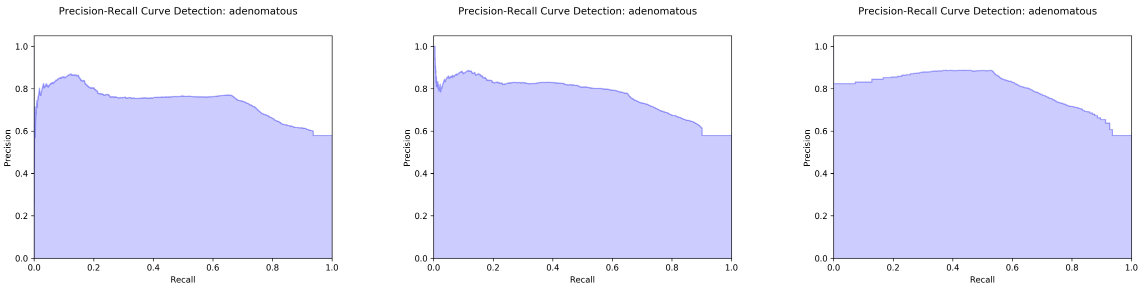


(a) DetNet (b) FasterRCNN (c) RefineDet

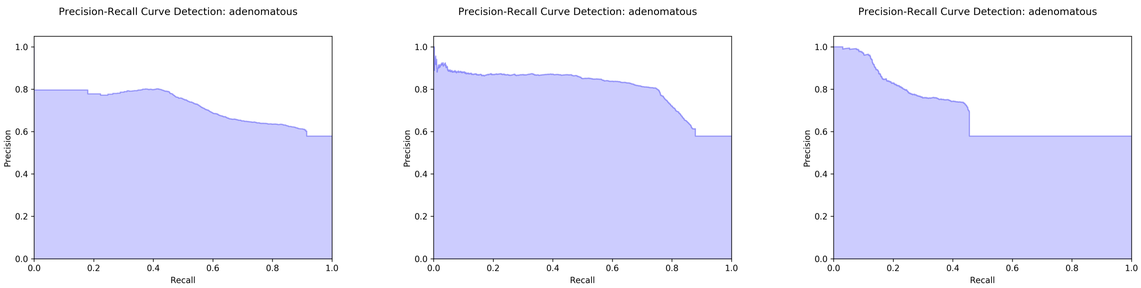


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.4: ROC Curve for 2-classes polyp detection: hyperplastic polyp

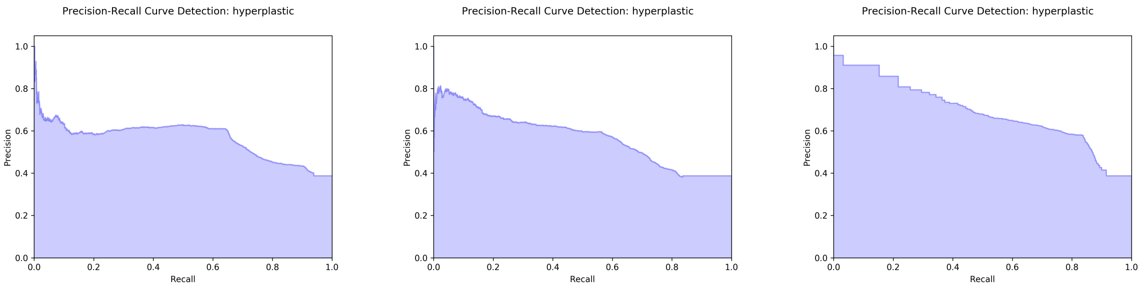


(a) DetNet (b) FasterRCNN (c) RefineDet

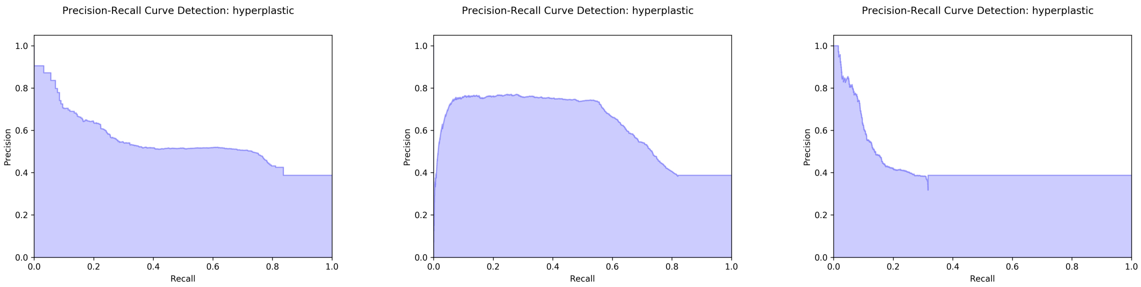


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.5: PR Curve for 2-classes polyp detection: adenomatous polyp



(a) DetNet (b) FasterRCNN (c) RefineDet



(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.6: PR Curve for 2-classes polyp detection: hyperplastic polyp

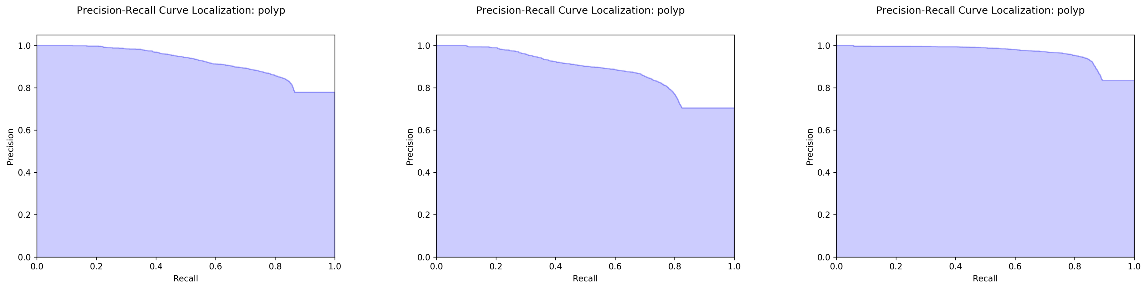
Here, we consider the task as 2-classes (adenomatous and hyperplastic) image classification task. Again, we do not consider the size and position prediction made by the models. We are interested in whether the models know what is the correct class for the polyp contained in the image frame (if it does contain a polyp). If the model predicts that there are multiple polyps in the image frame, we take the prediction with the highest confidence score as the final class prediction of the whole image frame. Table 5.2 shows different metric scores for the models. Figures in 5.3 and 5.4 show the ROC curve for each model prediction of adenomatous and hyperplastic, respectively. Figures in 5.5 and 5.6 show the Precision-Recall curve for each model prediction of adenomatous and hyperplastic, respectively.

We can see that SSD has better mean precision score compared to RefineDet and DetNet in this case. This means that more of those polyp frames predicted by RefineDet and DetNet are misclassified compared to SSD predictions. However, the mean recall scores for RefineDet and DetNet are still better than SSD. Overall, RefineDet has the highest mean average precision score, followed by SSD. DetNet has lower mean average precision score due to its lower mean precision score. Unsurprisingly, the combined performance has the best scores for all the metrics due to the way it was created.

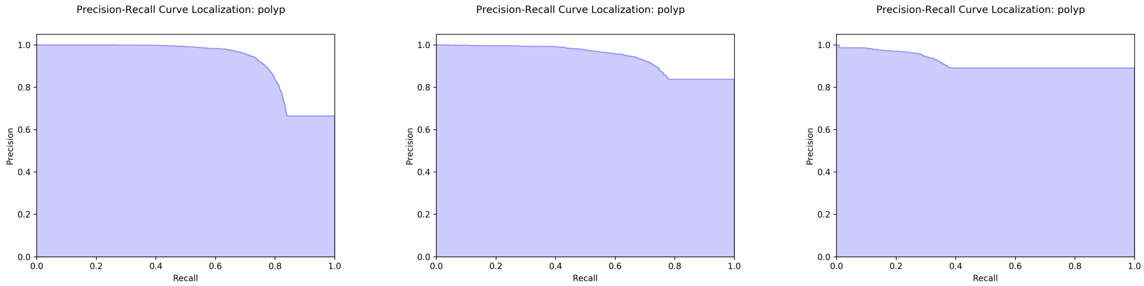
5.4 1-class Polyp Localization Task

Metrics	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
Prec	100	77.8	70.2	83.2	66.3	83.7	89.0
Rec	86.6	86.5	82.4	89.2	84.0	78.1	38.5
F1	92.8	81.9	75.8	86.1	74.1	80.8	53.8
AP	86.6	81.8	75.8	87.5	81.7	76.0	37.1

Table 5.3: Performance of each model under different metrics for 1-class polyp localization task (locating and classifying each prediction as polyp or non-polyp)



(a) DetNet (b) FasterRCNN (c) RefineDet



(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.7: PR Curve for 1-class polyp localization

Here, we consider the task as 1-class object detection task. Thus, we also consider the size and location prediction made by the models. This means that we use each bounding box prediction instead of each image prediction for comparison. We first compare the performances of the models considering just 1-class to see how well the models are able to localize the polyps, regardless of the class. If the models have good performances on this task but bad performances on the 2-class polyp localization task, then the models are having difficulty to classify the polyps. Table 5.3 shows different metric scores for the models. Figures in 5.7 show the Precision-Recall curve for each model.

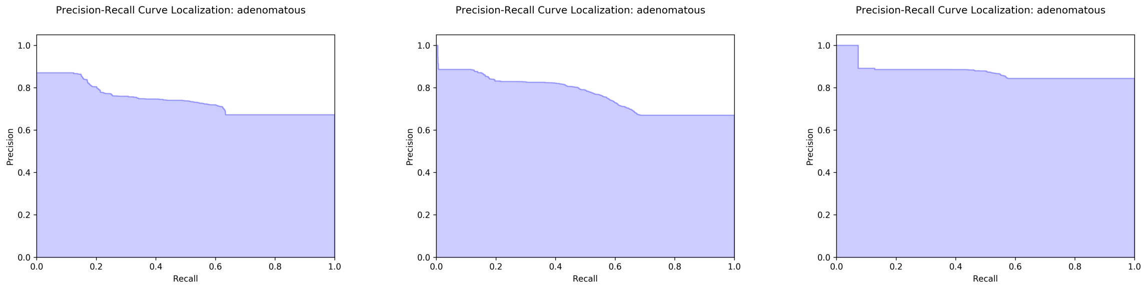
We can see that YOLOv3 has the highest precision score. However, this is because it makes fewer predictions compared to the other models. The recall and average precision scores for YOLOv3 also confirm this. So, this does not necessarily mean that YOLOv3 performed better than the other models.

Excluding YOLOv3, SSD has the best precision score followed by RefineDet. However, the overall recall score for RefineDet and DetNet are still higher than SSD. RetinaNet also has higher recall score than SSD. RefineDet has the highest average precision score, followed by DetNet and RetinaNet. These three models have good average precision scores (more than 80%) if we do not consider the class prediction.

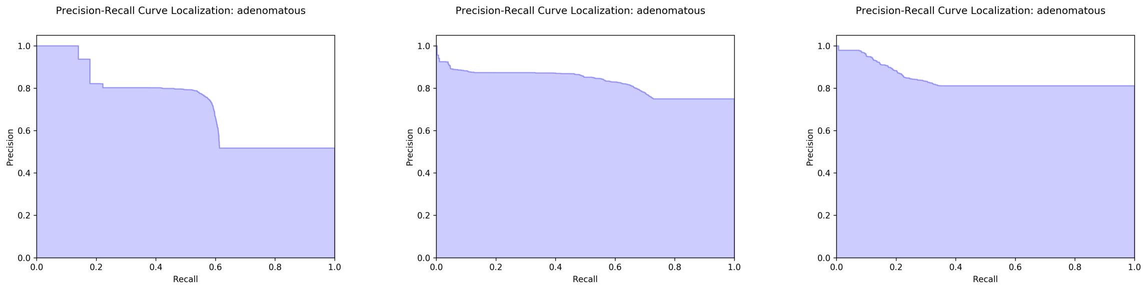
5.5 2-classes Polyp Localization Task

Metrics	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
Prec Ad	100	67.0	67.0	84.2	51.4	74.9	81.1
Prec Hy	100	43.0	41.0	48.6	40.4	53.3	52.3
mPrec	100	55.0	54.0	66.4	45.9	64.1	66.7
Rec Ad	84.9	63.4	68.6	57.5	61.3	72.8	34.9
Rec Hy	89.1	58.5	57.3	80.3	55.5	46.8	22.8
mRec	87.0	60.9	62.9	68.9	58.4	59.8	28.8
F1 Ad	91.8	65.2	67.8	68.4	55.9	73.8	48.8
F1 Hy	94.2	49.6	47.8	60.5	46.7	49.8	31.7
mF1	93.0	57.4	57.8	64.4	51.3	61.8	40.3
AP Ad	84.9	49.5	55.8	51.6	51.9	62.6	31.4
AP Hy	89.1	32.9	29.5	58.4	34.5	32.0	16.1
mAP	87.0	41.2	42.7	55.0	43.2	47.3	23.8

Table 5.4: Performance of each model under different metrics for 2-classes polyp localization task (locating and classifying each prediction as adenomatous, hyperplastic, or non-polyp)

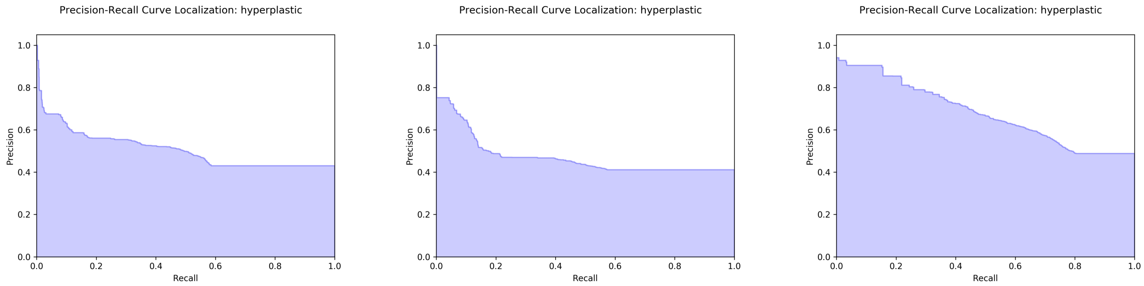


(a) DetNet (b) FasterRCNN (c) RefineDet

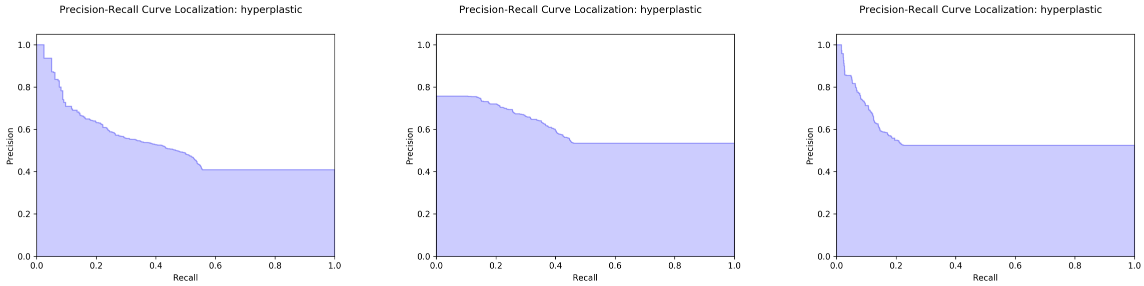


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.8: PR Curve for 2-classes polyp localization: adenomatous polyp



(a) DetNet (b) FasterRCNN (c) RefineDet



(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.9: PR Curve for 2-classes polyp localization: hyperplastic polyp

Here, we consider the task as 2-classes (adenomatous and hyperplastic) object detection task. This is the original computer vision task that is addressed in this comparative study. We take into account the class, size, and location prediction made by the models. Table 5.4 shows different metric scores for the models. Figures in 5.8 and 5.9 show the Precision-Recall curves for each model prediction of adenomatous and hyperplastic, respectively.

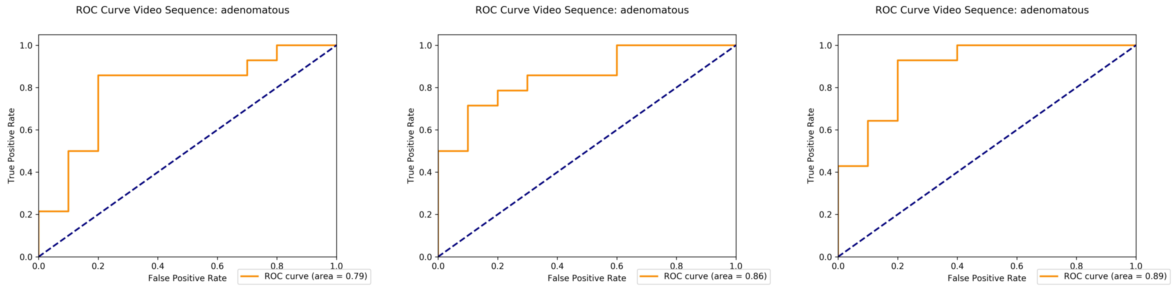
Similar to the previous analysis, YOLOv3 has the best precision due to fewer predictions. So, it is not particularly interesting to compare YOLOv3 precision scores. RefineDet has the best mean precision score, followed by SSD and DetNet (excluding YOLOv3). RefineDet also has the best mean recall score. The hyperplastic recall score for RefineDet is significantly higher compared to the other models. This means that RefineDet is very good at predicting correct hyperplastic polyps. However, the adenomatous recall score is lower compared to the other models (excluding YOLOv3). On the other hand, SSD has the highest adenomatous recall score and the lowest hyperplastic recall score compared to the other models (excluding YOLOv3). Surprisingly, FasterRCNN has the second best mean recall score. Finally, RefineDet has the highest mean average precision score, followed by SSD and RetinaNet.

The combined performance has good scores on most of the metric scores. It has a final mean average precision score of 87%. This means that it is possible to design and improve a model that has good performance for this particular dataset. The hyperplastic average precision score is high for the combined performance, despite the low scores for most of the models. This means that each model correctly predicted different hyperplastic frames. Thus, performance improvement on this dataset is possible.

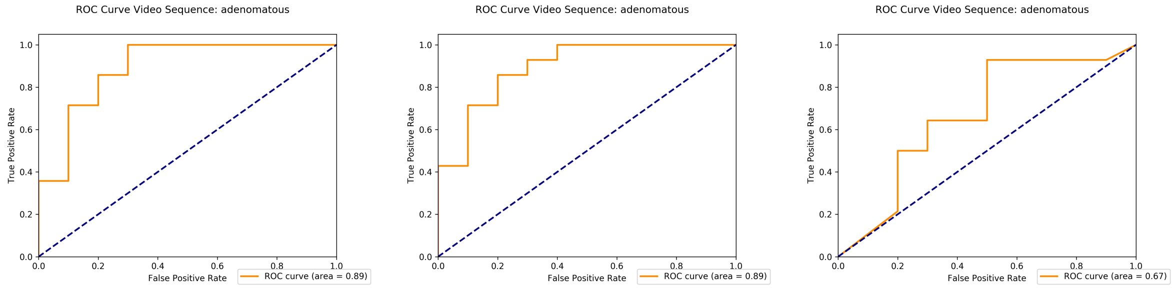
5.6 Video Sequence Classification Task

Metrics	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
Acc	1.0	79.1	79.1	62.5	83.3	83.3	58.3
Bal Acc	1.0	79.2	79.2	66.4	82.8	81.4	58.5
Prec Ad	1.0	84.6	84.6	85.7	85.7	81.2	72.7
Prec Hy	1.0	72.7	72.7	52.9	80.0	87.5	54.5
mPrec	1.0	78.6	78.6	69.3	82.8	84.3	63.6
Rec Ad	1.0	78.5	78.5	42.8	85.7	92.8	57.1
Rec Hy	1.0	80.0	80.0	90.0	80.0	70.0	60.0
mRec	1.0	79.2	79.2	66.4	82.8	81.4	58.5
F1 Ad	1.0	81.4	81.4	57.1	85.7	86.6	64.0
F1 Hy	1.0	76.1	76.1	66.6	80.0	77.7	57.1
mF1	1.0	78.8	78.8	61.9	82.8	82.2	60.5
AP Ad	1.0	84.3	90.9	91.7	91.2	91.9	68.2
AP Hy	1.0	74.9	82.4	89.6	90.1	89.2	57.8
mAP	1.0	79.6	86.7	90.6	90.6	90.5	63.0
AUC Ad	1.0	79.2	85.7	89.2	89.2	89.2	66.7
AUC Hy	1.0	79.2	85.7	89.2	89.2	89.2	69.2
mAUC	1.0	79.2	85.7	89.2	89.2	89.2	68.0

Table 5.5: Performance of each model under different metrics for video sequence classification task (classifying each video sequence as adenomatous, hyperplastic, or non-polyp)

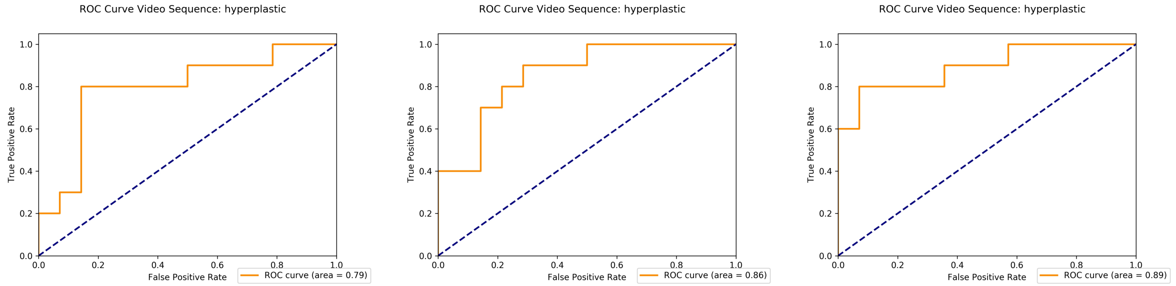


(a) DetNet (b) FasterRCNN (c) RefineDet

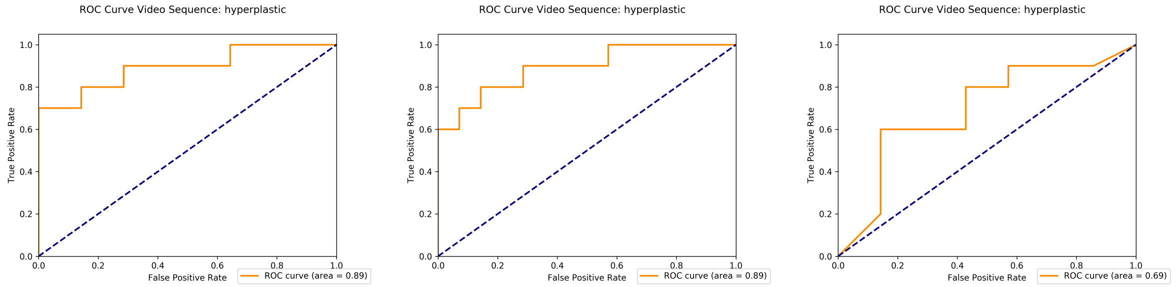


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.10: ROC Curve for 2-classes polyp detection per video sequence: adenomatous polyp

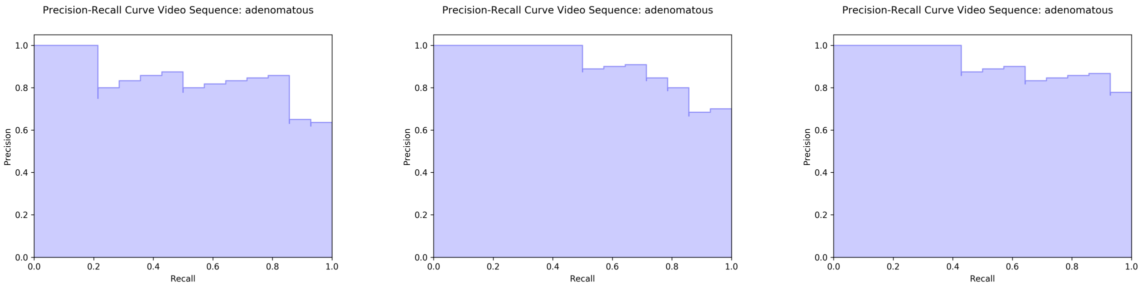


(a) DetNet (b) FasterRCNN (c) RefineDet

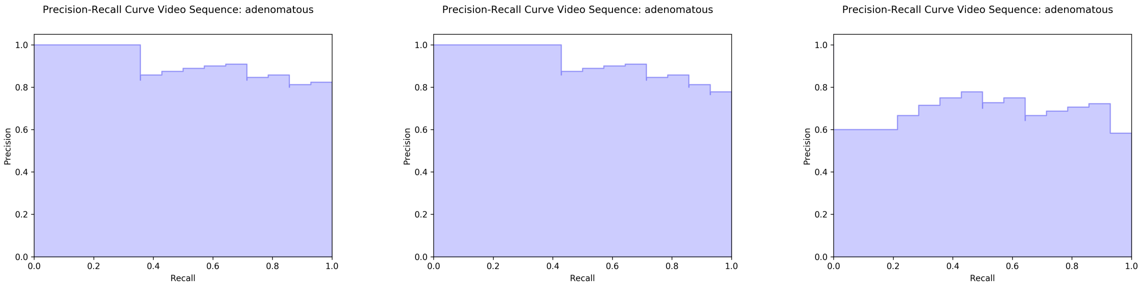


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.11: ROC Curve for 2-classes polyp detection per video sequence: hyperplastic polyp

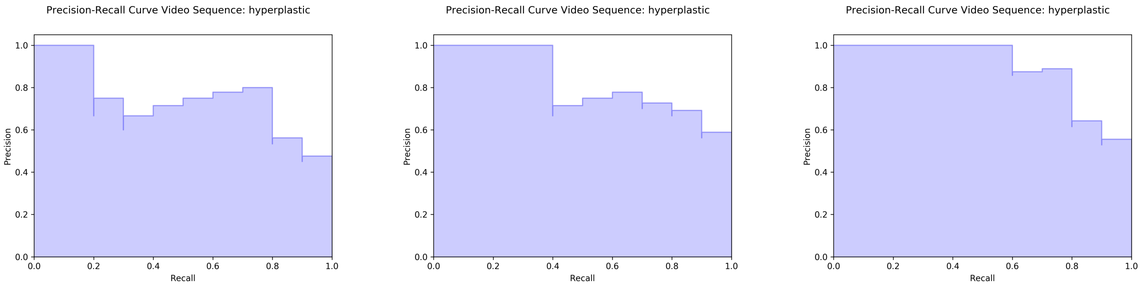


(a) DetNet (b) FasterRCNN (c) RefineDet

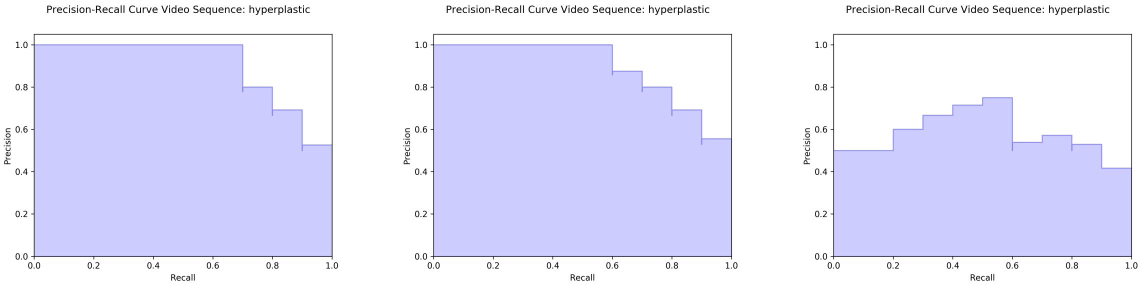


(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.12: PR Curve for 2-classes polyp detection per video sequence: adenomatous polyp

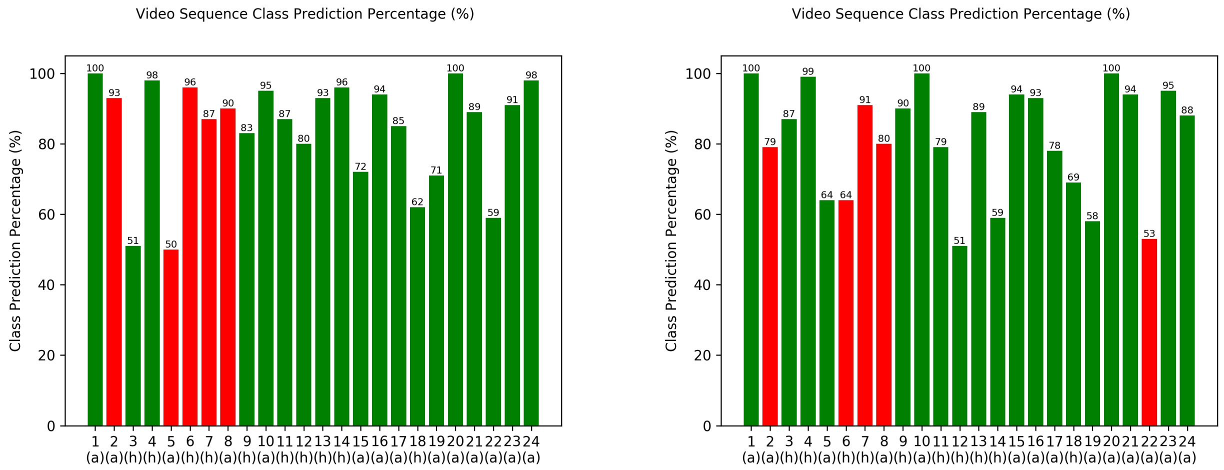


(a) DetNet (b) FasterRCNN (c) RefineDet



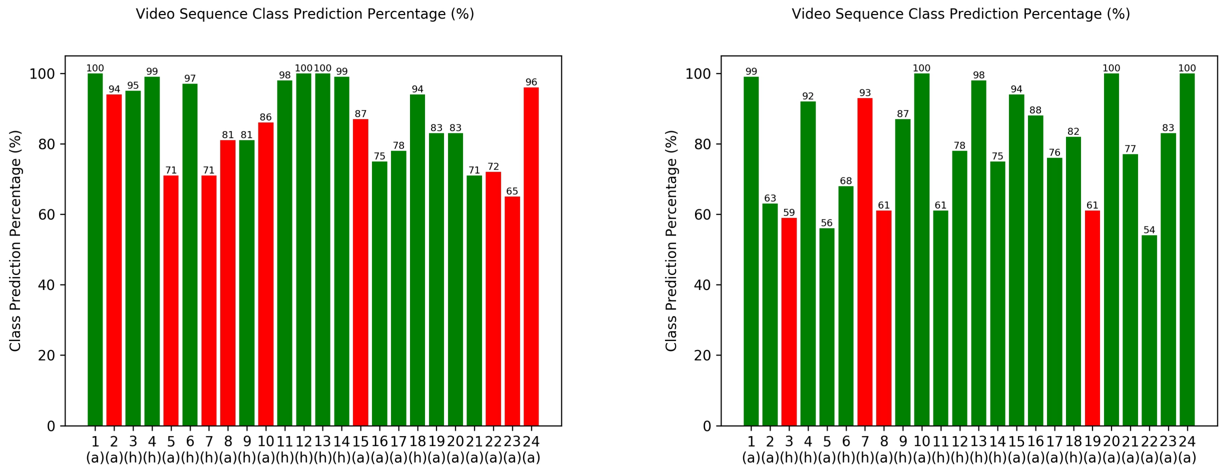
(d) RetinaNet (e) SSD (f) YOLOv3

Figure 5.13: PR Curve for 2-classes polyp detection per video sequence: hyperplastic polyp



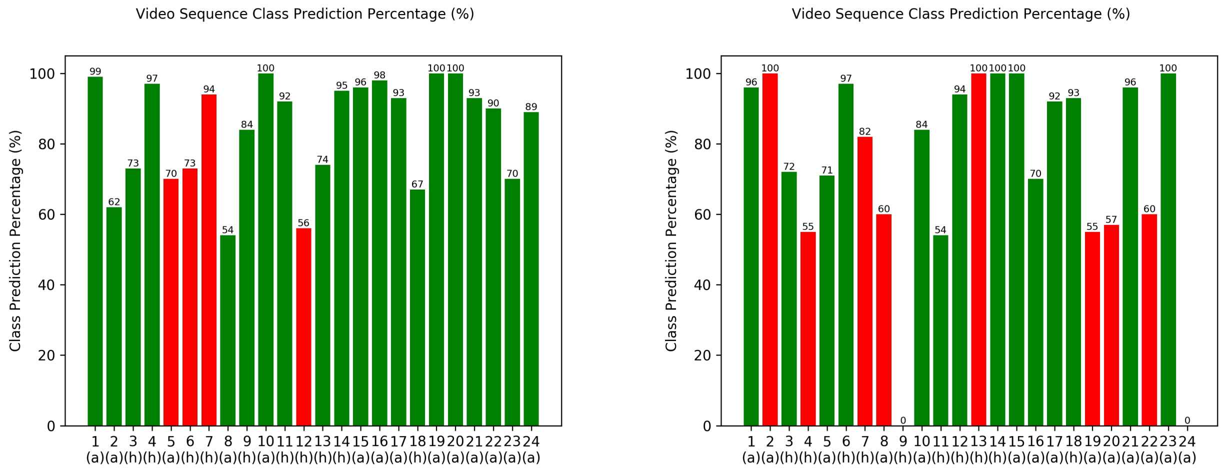
(a) DetNet

(b) FasterRCNN



(c) RefineDet

(d) RetinaNet



(e) SSD

(f) YOLOv3

Figure 5.14: Video sequence class prediction probability

Here, we consider the task as 2-classes (adenomatous and hyperplastic) classification task of the whole video sequence. Specifically, we are interested in the agreement of each model on the class prediction of the frames in a particular video sequence. This is useful when a physician wants to classify a polyp based on the whole video sequence. Sometimes, one or two frames are not sufficient to know the actual class of the polyp. Thus, looking at the agreement of class prediction across a sequence of frames might result in better class prediction for the polyp in the video sequence. We need to also note that we consider the prediction in image level instead of bounding box level. Table 5.5 shows different metric scores for the models. Figures in 5.10 and 5.11 show the ROC curve for each model prediction of adenomatous and hyperplastic, respectively. Figures in 5.12 and 5.13 show the Precision-Recall curve for each model prediction of adenomatous and hyperplastic, respectively. Figures in 5.14 show video sequence class prediction probability for each model. A red bar means that the model predicted the wrong class for the video sequence, while a green bar means correct class prediction. The y-axis shows the percentage of the final class prediction over all predicted frames. The x-axis shows the video sequence number; symbol a for adenomatous and h for hyperplastic.

We can see that SSD and RetinaNet have the highest accuracy scores, while RefineDet has lower accuracy score than the other models (excluding YOLOv3). This means that SSD and RetinaNet have better agreements in predicting correct polyp classification. RefineDet has a lower hyperplastic precision score because it misclassified more adenomatous video sequences. However, the overall hyperplastic recall score for RefineDet is higher than the other models. Another interesting observation is that RefineDet has a good mean average precision score, having a score similar to SSD and RetinaNet. This is because correct predictions have high probabilities, which results in a high mean average precision score. Even though the accuracy scores for DetNet and FasterRCNN are higher than RefineDet, these models have lower mean average precision than RefineDet because they have misclassified predictions with high probability as we can see from the Precision-Recall curves for the models.

DetNet, FasterRCNN, and RetinaNet are not very consistent in predicting the polyp class in some of the video sequences, shown by prediction percentages that are close to 50% in 5.14. RefineDet and SSD, on the other hand, are relatively more consistent in predicting the polyp class in a video sequence. YOLOv3 performed the worse, having many misclassifications (with 1 video sequence having no prediction at all) and prediction percentages close to 50 %.

5.7 Image Frame Prediction Analysis

After analyzing predictions made by the models based on previously discussed computer vision tasks, let us now analyze the predictions based on image frame category. We compare frames containing correct prediction, frames containing misclassified prediction, frames containing mislocalized prediction, frames containing multiple predictions, polyp frames containing no prediction, and non-polyp frames containing a prediction.

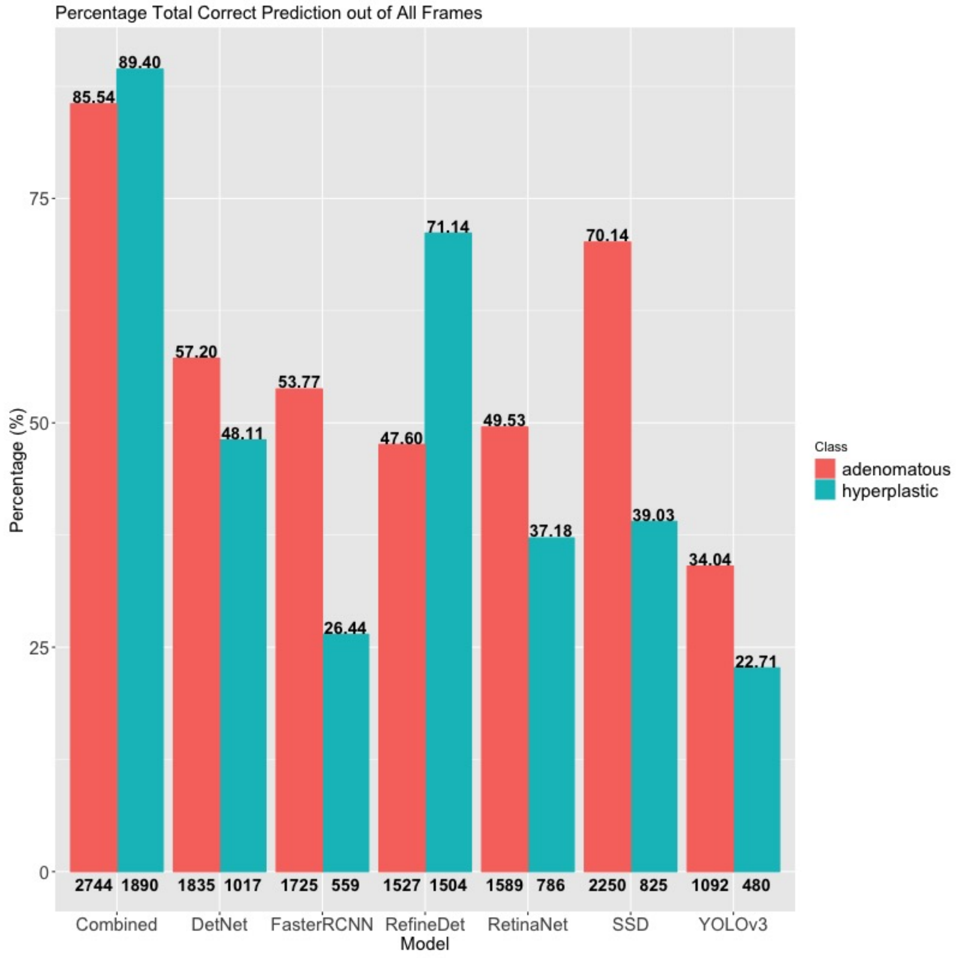


Figure 5.15: Percentage of total correct prediction for each model

Figure 5.15 shows the percentage of correctly predicted frames over all number of frames for each class. Here, correct prediction means that the model predicts the correct class, size, location, and number of polyps in the image frame. This shows that the model successfully predicts the image frame according to the ground truth with no error.

Most of the models successfully predicted around 50% of all frames with adenomatous class, with SSD having the best performance of correctly predicting around 70% of all frames with adenomatous class. The models have bad performances on predicting hyperplastic frames at around 22% - 48%, with the exception of RefineDet. In order to have good performances, we need to increase the models' performances on the frames with hyperplastic class. RefineDet has unusually good performance on hyperplastic class at 71%, which explains the high mean average precision score it has.

We can see that the combined performance has a high percentage of correctly predicted frames. This means that the correct prediction frames are spread over different frames, which when combined covers a large portion of the image frames. Thus, it is possible to improve a model's performance to achieve at least this performance given the current dataset.

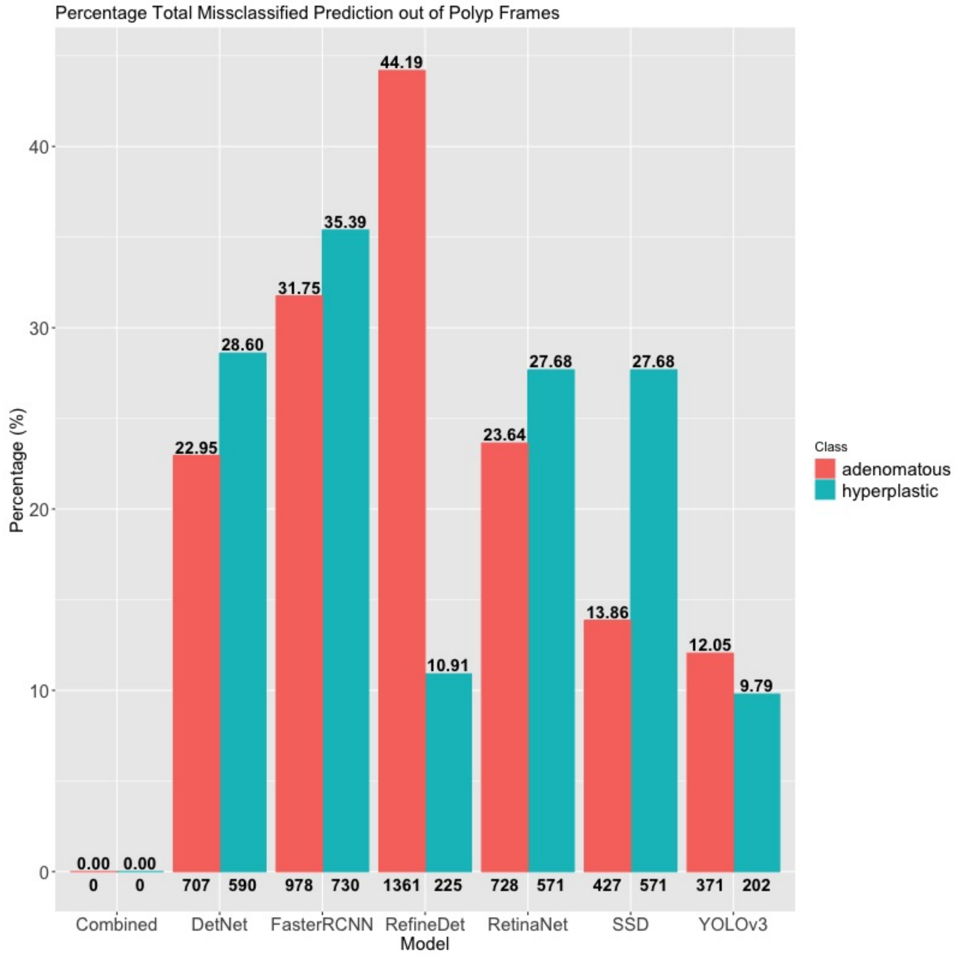


Figure 5.16: Percentage of total misclassified prediction for each model

Figure 5.16 shows the percentage of polyp frames with misclassified prediction over all polyp frames for each class. Here, misclassified prediction means that the model predicts the correct size and location, but incorrect class. This shows that the model is actually able to find the size and location of the polyp but having difficulty in classifying the polyp.

We can see that RefineDet has the highest misclassified prediction on adenomatous frames.

RefineDet predicts more hyperplastic class than adenomatous class, resulting in a large proportion of the adenomatous ground truth to be misclassified as hyperplastic by the model. Thus, it has a high hyperplastic recall score, but a low adenomatous precision score. The other models mostly failed to predict the correct class; predicting adenomatous class for hyperplastic polyp.

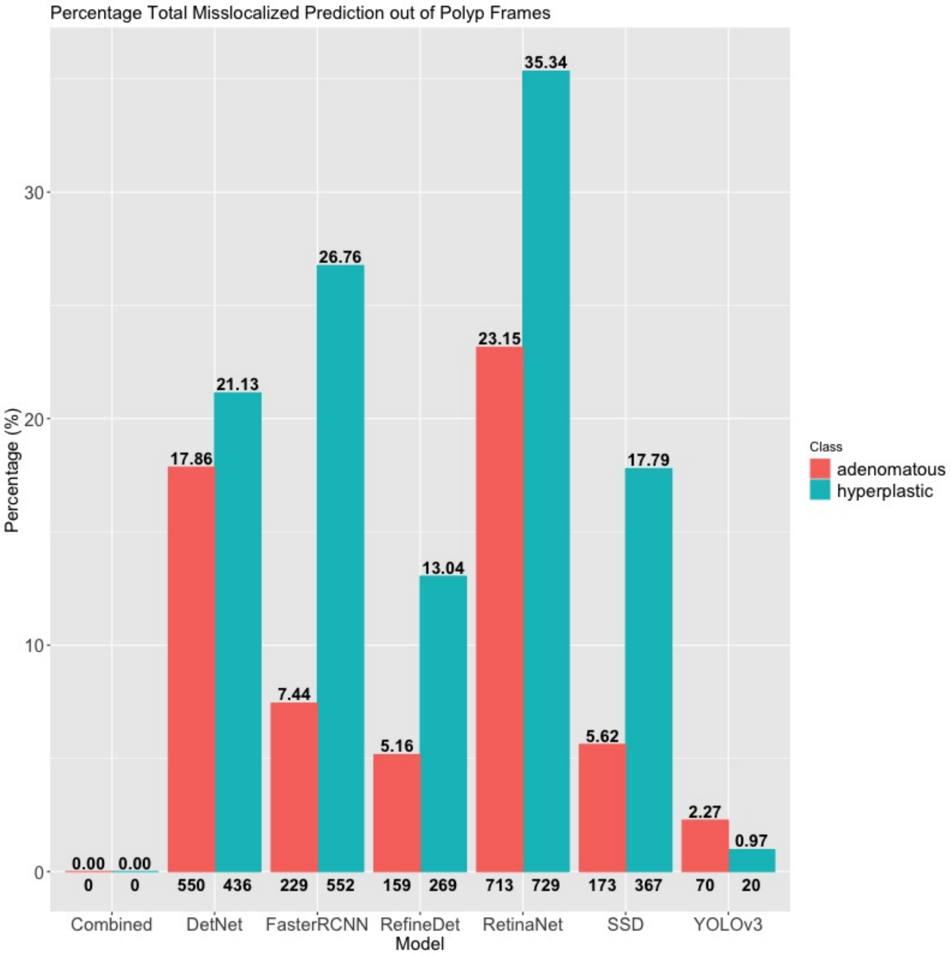


Figure 5.17: Percentage of total mislocalized prediction for each model

Figure 5.17 shows the percentage of polyp frames with mislocalized prediction over all polyp frames for each class. Here, mislocalized prediction means that the model makes a prediction at the wrong size and location, regardless of the class. So, the model detects that there is a polyp at a place that does not actually contain a polyp. Note that multiple predictions at the same correct location and correct size do not count as mislocalized predictions.

We can see that RetinaNet has the worst performance at localizing the prediction. This means that when RetinaNet makes a prediction, that prediction is mostly at the wrong location or having the wrong size. The other models are also having difficulty at making localization prediction for hyperplastic frames. This might be due to the shape and texture of hyperplastic polyp that is harder to distinguish from the colonic wall compared to adenomatous polyp.

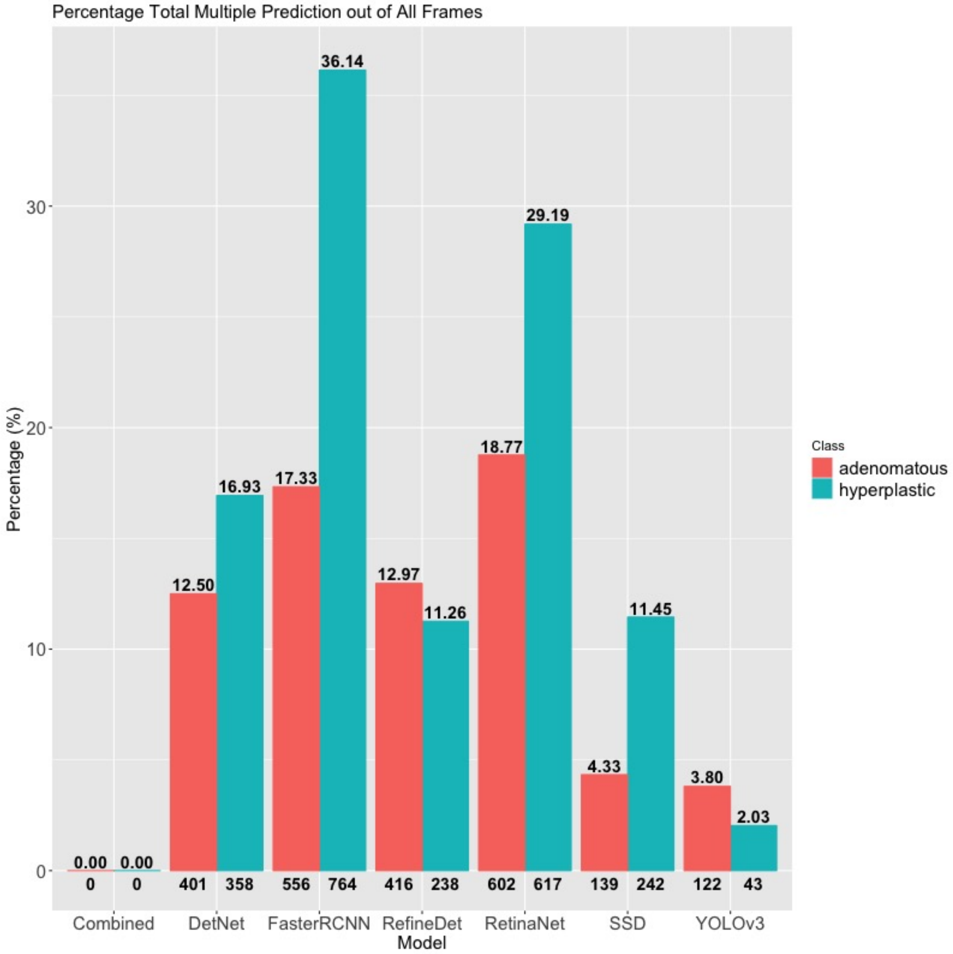


Figure 5.18: Percentage of total multiple predictions for each model

Figure 5.18 shows the percentage of polyp frames with multiple predictions over all frames for each class. Here, multiple predictions mean that the model predicts that there are multiple polyps in the image frame. Based on the way we created the dataset, there should be no image frame containing multiple polyps. Thus, if the model makes multiple predictions for the image frame, then we know that there are false positive predictions in the image. This can affect the mislocalized prediction if the multiple predictions are located at the wrong location or having the wrong size. The confidence score threshold we set for the post-processing step affects this result. Lower confidence score threshold introduces more false positive to the prediction results. Thus, we should not choose a confidence score threshold that is too low as it will not be usable in the real world.

FasterRCNN makes more multiple predictions on the hyperplastic frames compared to the other models, followed by RetinaNet. This is particularly interesting since the confidence score threshold for FasterRCNN was set to be 0.5, which is relatively high. The confidence score threshold for RetinaNet was set to be 0.05, so it does make more multiple predictions due to the threshold. FasterRCNN, on the other hand, makes multiple predictions despite the high confidence score threshold. This means that FasterRCNN is having difficulty in predicting hyperplastic polyp and distinguishing it from the colonic wall.

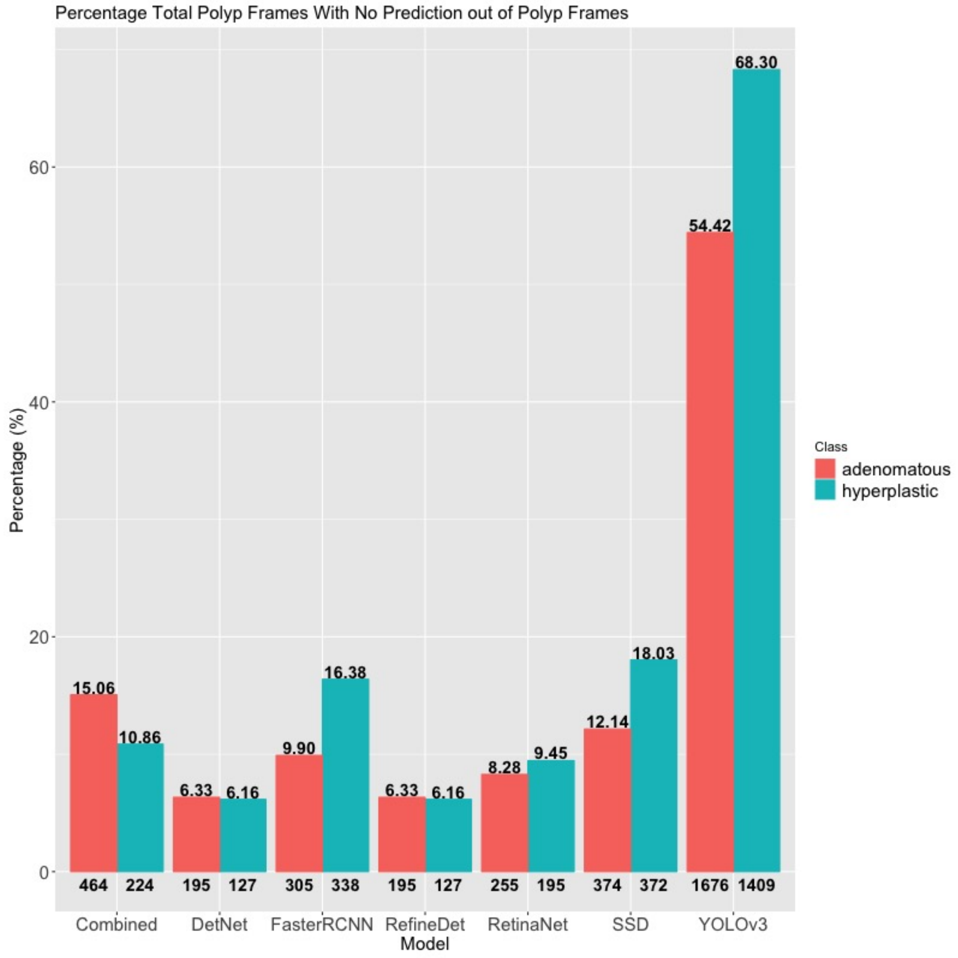


Figure 5.19: Percentage of total polyp frames without prediction for each model

Figure 5.19 shows the percentage of polyp frames with no prediction over all polyp frames for each class. Here, it means that the model predicts that the image frame contains no polyp while it actually contains a polyp. So, it shows how good the models are at not missing polyp in the image frame.

YOLOv3 has a very high percentage of this category due to the low number of high confidence scores for its predictions. With a confidence score threshold of 0.3 to minimize false positive predictions, it removes large portions of the correct predictions with low confidence scores. Thus, it has a very high percentage of this category. The other models have around 6% - 18% of this category, which is relatively small compared to the misclassification percentage. So, the models are mostly having difficulty in distinguishing the two polyp classes from each other. RefineDet and

DetNet are performing the best, having the smallest number of frames in this category.

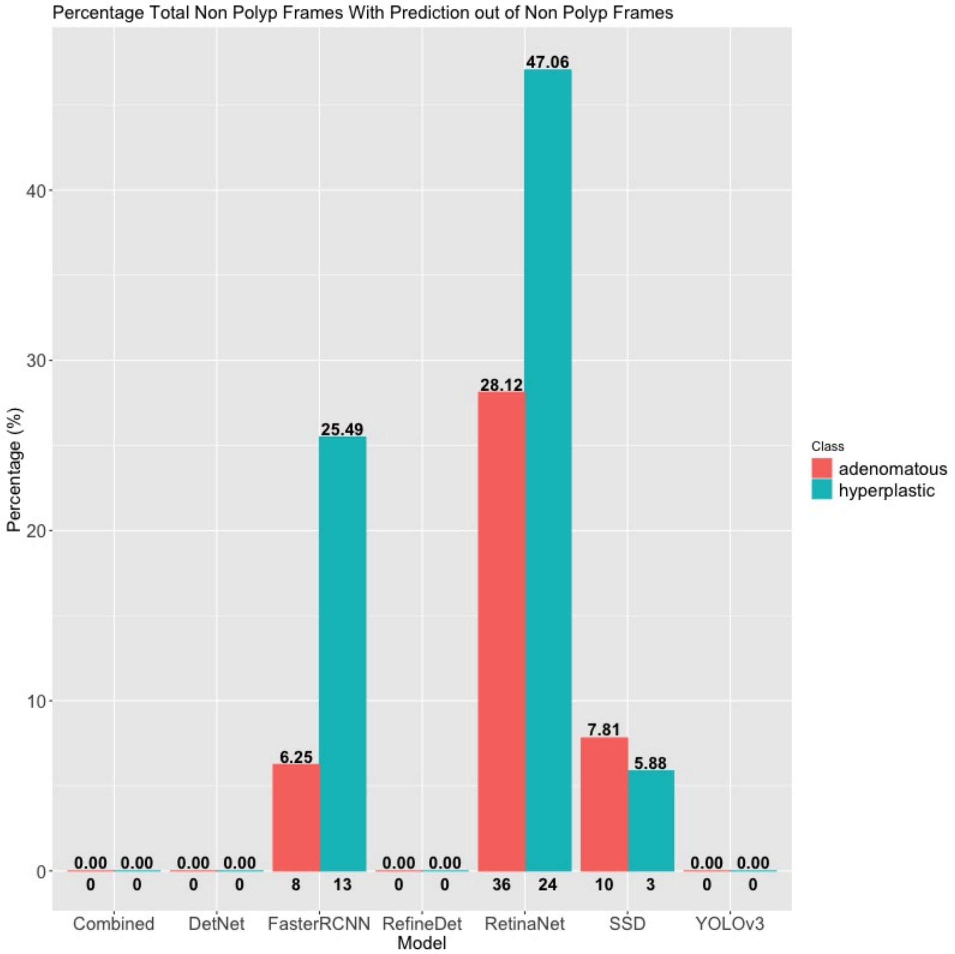
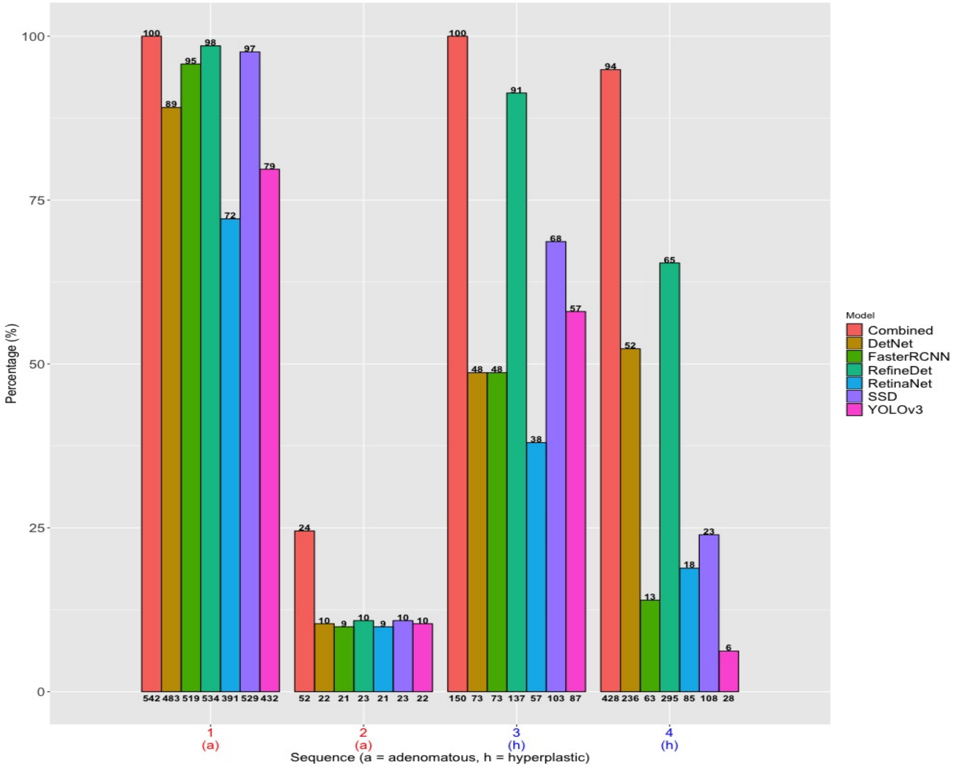


Figure 5.20: Percentage of total non polyp frames with prediction for each model

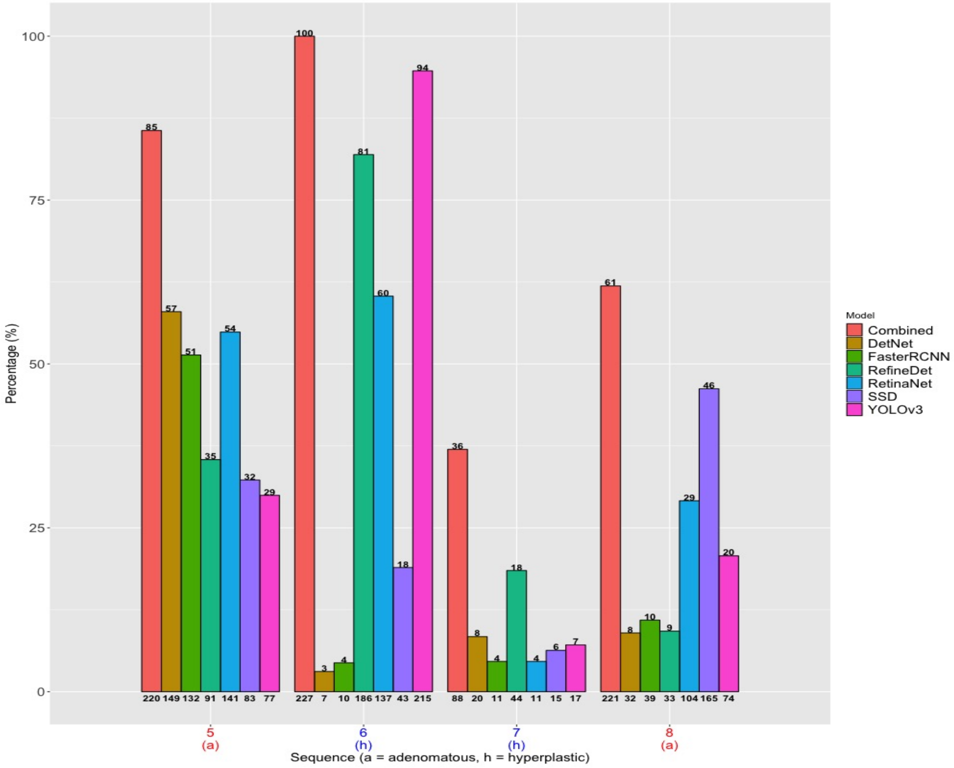
Figure 5.20 shows the percentage of non-polyp frames with prediction over all non-polyp frames. Here, it means that the model predicts that there is a polyp in the image frame while it actually contains no polyp. So, it shows how much the models make false positive predictions on non-polyp frames.

We can see that three of the models know when there is no polyp in the image frame. RetinaNet has the worst performance, having more than 20% of the total non-polyp frames in this category. FasterRCNN also has a relatively large percentage on the hyperplastic frames.

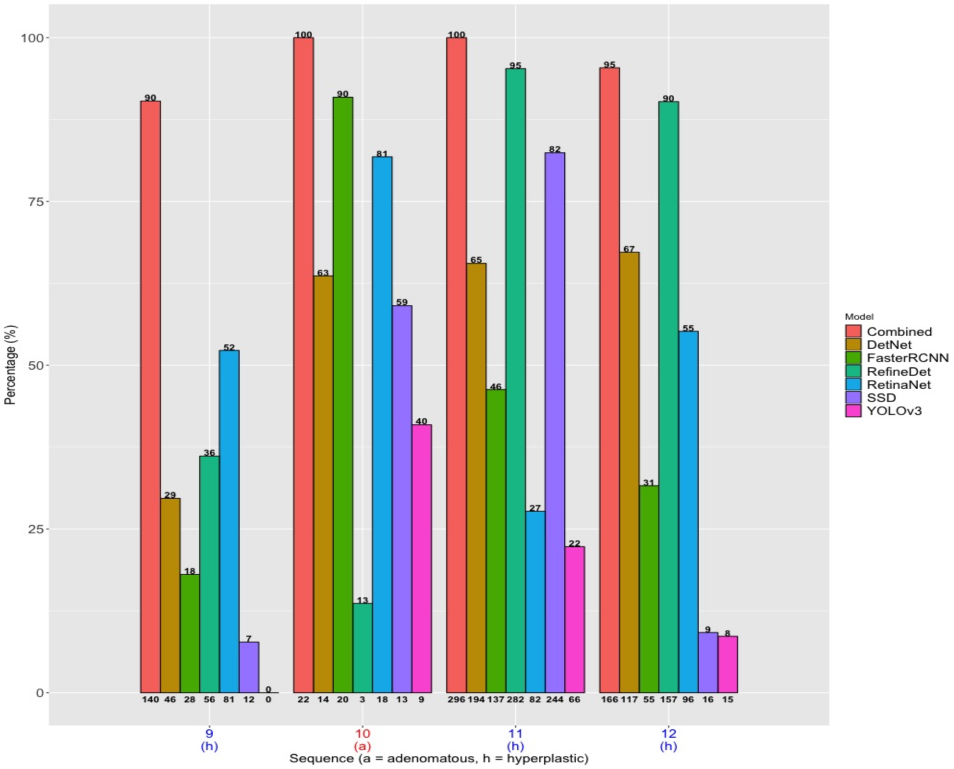
5.8 Video Sequence Analysis



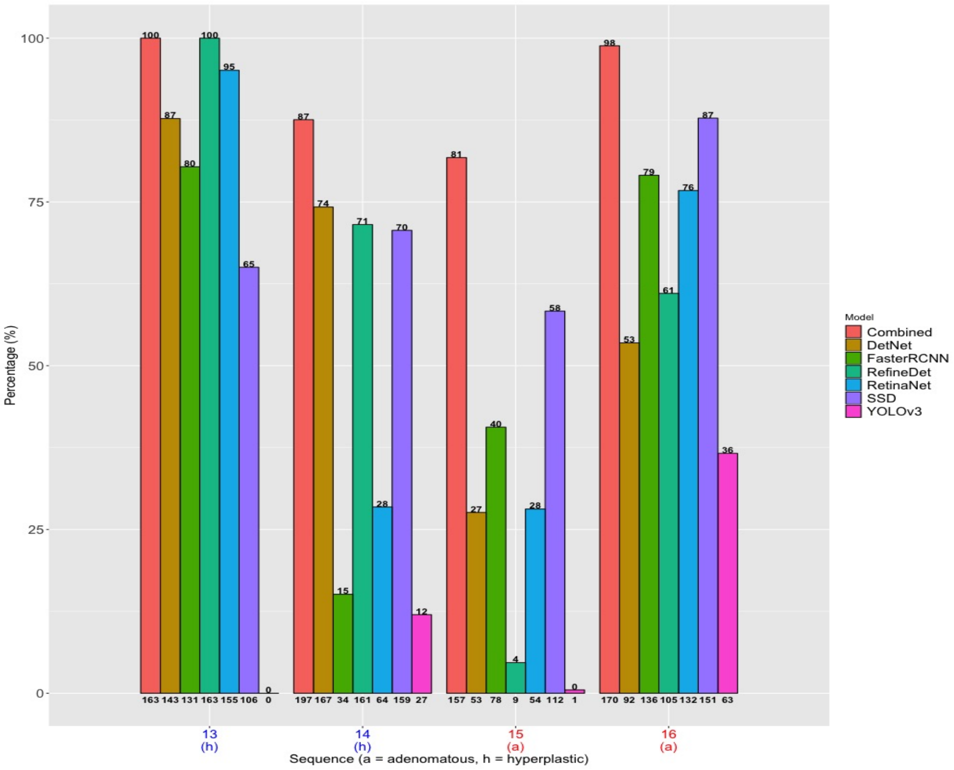
(a) Video sequence 1 - 4



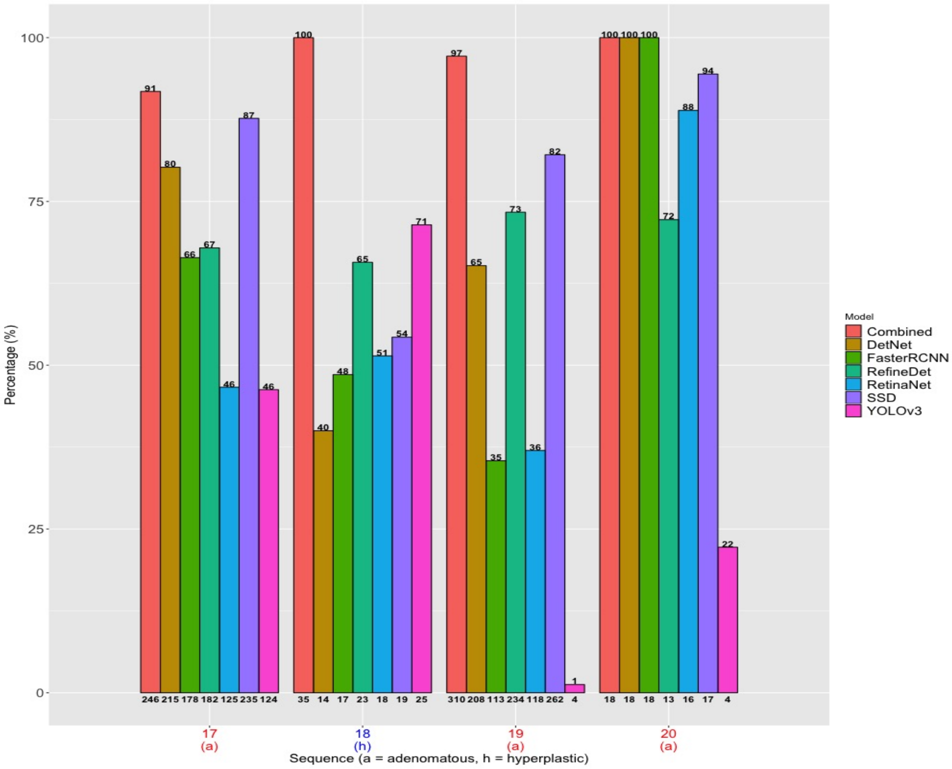
(b) Video sequence 5 - 8



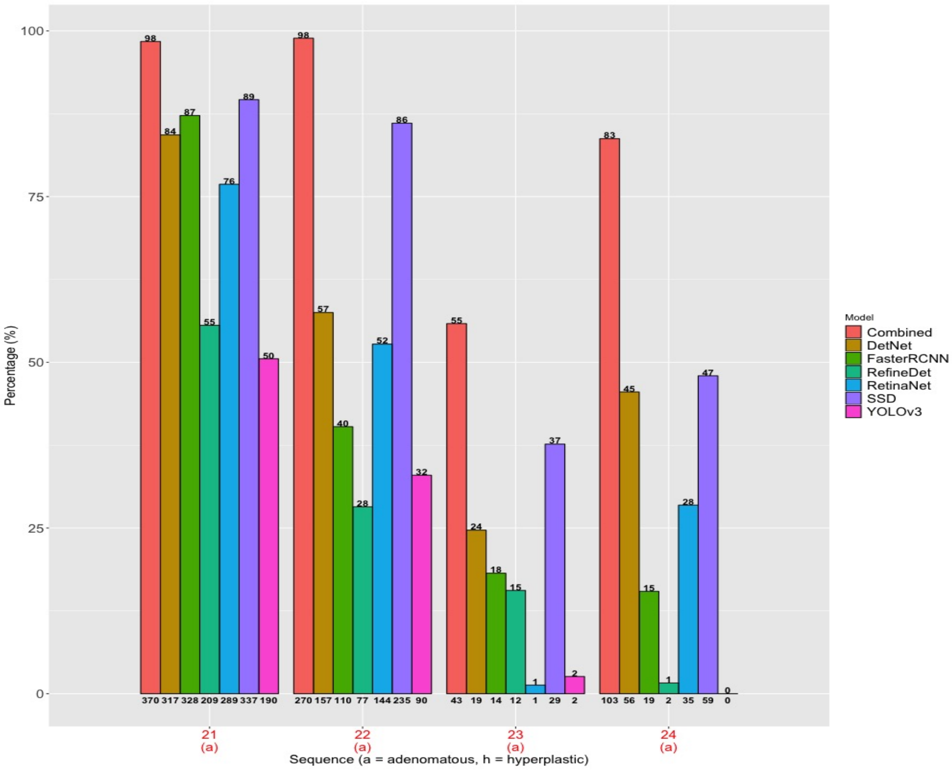
(c) Video sequence 9 - 12



(d) Video sequence 13 - 16



(e) Video sequence 17 - 20



(f) Video sequence 21 - 24

Figure 5.19: Percentage of correctly predicted frames per video sequence for each model. Numbers on top of the bars show the percentage values, while numbers below the bars show the actual number of frames

In this section, we analyze the agreement for the models regarding the video sequences in the dataset. We aim to see if the models mostly agree that a particular video sequence is a difficult sequence or a relatively easy sequence for the object detection task. Figures in 5.19 show the percentages of correct predictions made by the models in each video sequence. The symbol *a* in x-axis means that the video sequence is of adenomatous class, while the symbol *h* means that the video sequence is of hyperplastic class.

As we can see, the percentages of correct prediction vary depending on the model and the video sequence. Video sequences 6, 10, 11, 12, 14, and 22 have high variations between the models. However, all the models agree that video sequences 2, 7, 8, 23, and 24 are difficult sequences because none of the models correctly predict more than 50% of the image frames in these video sequences. All the models, excluding YOLOv3, also agree that video sequences 1, 13, 16, 20, and 21 are relatively easy sequences because all the models correctly predict more than 50% of the image frames in these video sequences.

RefineDet has the best performance on most of the hyperplastic sequences, which is in line with what we have observed so far. YOLOv3 surprisingly has the best performance on sequence 6, surpassing all the other models. Another interesting observation is that the combined performance on sequences like 2, 4, 5, 7, 8, 9, 18, 23, and 24 are relatively high compared to each individual model performance. This means that the models correctly predicted different frames of the video sequences, which when combined cover large portions of the video sequences. Thus, performance improvement is possible for the models.

5.9 Image Frame Prediction Summary

In this section, we present summaries regarding the prediction results made by the models. These summaries categorize the image frames according to the previously discussed analysis.

- Table 5.6 shows the polyp class, number of frames, number of polyp frames, and number of non-polyp frames in each video sequence

- Table 5.7 shows correctly predicted frames. SSD has the highest total of correctly predicted frames, followed by RefineDet. The combined performance has $4634/5322 = 87\%$ of all frames correctly predicted, which is very good

- Table 5.8 shows misclassified prediction frames. FasterRCNN has the highest misclassification followed by RefineDet as previous observation

- Table 5.9 shows mislocalized prediction frames. RetinaNet has the highest mislocalized prediction frames followed by DetNet

- Table 5.10 shows multiple prediction frames. FasterRCNN has the highest multiple predictions followed by RefineDet

- Table 5.11 shows polyp frames with no prediction. YOLOv3 has the highest polyp frames with no prediction followed by SSD

- Table 5.12 shows non-polyp frames with prediction. RetinaNet has the highest non-polyp frames with prediction followed by FasterRCNN

Seq	Class	Number of Frames	Polyp Frames	Non Polyp Frames
1	adenomatous	542	521	21
2	adenomatous	212	190	22
3	hyperplastic	150	143	7
4	hyperplastic	451	425	26
5	adenomatous	257	213	44
6	hyperplastic	227	227	0
7	hyperplastic	238	234	4
8	adenomatous	357	355	2
9	hyperplastic	155	155	0
10	adenomatous	22	22	0
11	hyperplastic	296	292	4
12	hyperplastic	174	174	0
13	hyperplastic	163	163	0
14	hyperplastic	225	215	10
15	adenomatous	192	192	0
16	adenomatous	172	172	0
17	adenomatous	268	268	0
18	hyperplastic	35	35	0
19	adenomatous	319	319	0
20	adenomatous	18	18	0
21	adenomatous	376	375	1
22	adenomatous	273	235	38
23	adenomatous	77	77	0
24	adenomatous	123	123	0
	Total	5322	5143	179

Table 5.6: Summary for number of frames and classification of each video sequence

Seq	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
1	542	483	519	534	391	529	432
2	52	22	21	23	21	23	22
3	150	73	73	137	57	103	87
4	428	236	63	295	85	108	28
5	220	149	132	91	141	83	77
6	227	7	10	186	137	43	215
7	88	20	11	44	11	15	17
8	221	32	39	33	104	165	74
9	140	46	28	56	81	12	0
10	22	14	20	3	18	13	9
11	296	194	137	282	82	244	66
12	166	117	55	157	96	16	15
13	163	143	131	163	155	106	0
14	197	167	34	161	64	159	27
15	157	53	78	9	54	112	1
16	170	92	136	105	132	151	63
17	246	215	178	182	125	235	124
18	35	14	17	23	18	19	25
19	310	208	113	234	118	262	4
20	18	18	18	13	16	17	4
21	370	317	328	209	289	337	190
22	270	157	110	77	144	235	90
23	43	19	14	12	1	29	2
24	103	56	19	2	35	59	0
Total	4634	2852	2284	3031	2375	3075	1572

Table 5.7: Number of correctly predicted frames. Here, correct prediction means correct classification, localization, and number of prediction each frame

Seq	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
1	0	1	0	5	0	0	22
2	0	46	86	62	16	17	1
3	0	69	62	11	84	45	39
4	0	4	6	0	22	0	0
5	0	105	114	153	84	145	12
6	0	219	216	24	72	183	2
7	0	200	207	166	207	203	70
8	0	268	273	287	172	143	111
9	0	0	0	12	8	5	0
10	0	1	0	19	0	6	10
11	0	37	75	6	112	36	59
12	0	33	59	2	31	29	2
13	0	9	30	0	2	47	25
14	0	6	59	0	27	10	0
15	0	25	17	141	1	1	0
16	0	9	25	61	17	4	43
17	0	35	77	76	59	19	15
18	0	13	16	4	6	13	5
19	0	89	192	83	196	11	5
20	0	0	0	5	0	1	10
21	0	32	34	160	78	26	6
22	0	94	160	196	105	33	136
23	0	1	0	39	0	12	0
24	0	1	0	74	0	9	0
Total	0	1297	1708	1586	1299	998	573

Table 5.8: Number of misclassified prediction frames. Here, misclassified prediction means frame containing prediction at the correct location but wrong classification prediction

Seq	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
1	0	45	20	1	149	4	7
2	0	109	48	48	113	70	0
3	0	7	21	2	15	0	0
4	0	180	347	116	304	277	7
5	0	5	8	12	34	11	15
6	0	36	5	19	29	3	3
7	0	35	20	29	72	33	4
8	0	46	11	10	68	23	30
9	0	54	35	73	39	25	0
10	0	8	2	1	4	4	0
11	0	60	66	6	158	7	5
12	0	27	42	6	49	8	0
13	0	8	2	0	3	4	1
14	0	19	12	11	46	7	0
15	0	56	59	21	96	13	0
16	0	70	8	6	26	13	0
17	0	9	4	5	78	2	1
18	0	10	2	7	14	3	0
19	0	42	1	2	6	1	0
20	0	0	0	0	2	0	0
21	0	33	11	6	21	5	1
22	0	31	8	0	38	5	16
23	0	38	32	14	30	3	0
24	0	58	17	33	48	19	0
Total	0	986	781	428	1442	540	90

Table 5.9: Number of misslocalized prediction frames. Here, misslocalized prediction means frame containing prediction at the wrong location (regardless of class prediction)

Seq	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
1	0	59	19	6	147	2	16
2	0	54	20	13	74	15	0
3	0	7	65	7	13	19	23
4	0	134	317	82	250	102	1
5	0	3	78	30	29	25	1
6	0	36	134	37	29	24	2
7	0	29	29	38	57	18	4
8	0	19	60	43	55	29	19
9	0	38	12	51	34	9	0
10	0	8	2	1	4	9	8
11	0	64	75	9	133	32	7
12	0	27	56	3	47	8	1
13	0	7	31	0	1	14	0
14	0	6	34	3	39	8	0
15	0	36	54	26	74	1	0
16	0	69	26	36	24	10	19
17	0	7	36	31	74	4	7
18	0	10	11	8	14	8	5
19	0	42	119	43	6	12	2
20	0	0	0	5	2	1	3
21	0	30	35	88	19	6	2
22	0	31	89	72	34	22	45
23	0	20	15	6	20	1	0
24	0	23	3	16	40	2	0
Total	0	759	1320	654	1219	381	165

Table 5.10: Number of multiple predictions frames. Here, multiple predictions means frame containing more than one predictions

Seq	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
1	0	0	1	2	0	9	76
2	160	59	75	89	74	109	189
3	0	1	1	0	0	2	23
4	23	31	39	39	54	66	416
5	37	0	4	1	2	18	153
6	0	0	0	0	0	0	5
7	150	1	10	0	1	2	147
8	136	23	39	30	40	28	143
9	15	55	92	24	29	116	155
10	0	0	0	0	0	0	3
11	0	3	30	2	13	9	166
12	8	2	44	9	4	121	157
13	0	0	0	0	3	6	137
14	28	34	120	52	91	49	198
15	35	73	40	27	42	66	191
16	2	2	3	3	0	7	65
17	22	10	9	6	11	12	128
18	0	0	2	1	0	1	5
19	9	0	13	1	1	45	310
20	0	0	0	0	0	0	4
21	6	0	3	3	0	10	179
22	3	0	0	0	0	1	37
23	34	19	31	13	46	33	75
24	20	9	87	20	39	36	123
Total	688	322	643	322	450	746	3085

Table 5.11: Number of polyp frames without prediction. Here, it means frame containing polyp but the model predicted that it does not contain polyp

Seq	Combined	DetNet	FasterRCNN	RefineDet	RetinaNet	SSD	YOLOv3
1	0	0	3	0	5	1	0
2	0	0	3	0	17	5	0
3	0	0	5	0	3	0	0
4	0	0	5	0	13	3	0
5	0	0	0	0	9	2	0
6	0	0	0	0	0	0	0
7	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	2	0	1	0	0
12	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0
14	0	0	1	0	6	0	0
15	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0
22	0	0	2	0	5	2	0
23	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0
Total	0	0	21	0	60	13	0

Table 5.12: Number of non-polyp frames with prediction. Here, it means frame containing no polyp but the model predicted that it contains polyp

Chapter 6

Conclusions

From the confidence score analysis, we found that YOLOv3 has mostly low confidence scores for correct predictions. This results in its low performances throughout our analysis. RetinaNet has the smallest number of false positive predictions compared to the other models, which might be the benefit of using focal loss. RefineDet has the highest number of correct predictions with confidence score ≥ 0.5. This results in its good performances throughout our analysis.

We also found some interesting observations from each previously discussed computer vision task. We found that RefineDet and DetNet have the best performances in predicting whether there is a polyp or not in the image frame. These two models have 100% precision score and 93.7% recall score for this particular task. In 2-classes image classification task, SSD has a higher mean precision score than RefineDet and DetNet but a lower overall mean recall score than these two models. Again, RefineDet has the best mean average precision score for this particular task.

In 1-class object detection task, SSD also has a higher precision score but a lower recall score compared to RefineDet, DetNet, RetinaNet. When we do not consider the class prediction, RefineDet, DetNet, and RetinaNet have good performances with average precision scores of $\geq$ 80%. However, once we start to consider the class prediction as in the 2-classes polyp localization task, the metric scores decrease due to classification error. The models, excluding YOLOv3, have mean average precision scores of around 40% - 55% because of poor performances on the hyperplastic frames. RefineDet's good performance on the hyperplastic frames (with a hyperplastic recall score of 80.3 %) results in the highest mean average precision compared to the other models. Thus, to improve the performances of the models, we need to improve their performances on the hyperplastic set. The resulting good combined performance also supports this finding that good performance on this dataset is possible.

We also analyzed how consistent the models are at predicting the correct polyp classification of image frames containing the same polyp from different viewpoints. We looked at the class prediction percentage over all image frames per video sequence. We found that the models are sometimes not sure about the correct class prediction for the same polyp, shown by class prediction percentage close to 50%. However, sometimes the models are also very consistent at predicting incorrect class prediction for the polyp. RefineDet and SSD are more consistent at making class prediction of the whole video sequences (regardless of correct or wrong predictions) because they have fewer predictions close to 50% compared to RefineDet, FasterRCNN, and RetinaNet.

From the image frame prediction analysis, we found that RefineDet outperformed all the other models (which have bad performances) in correctly predicting hyperplastic frames. We also found that most of the bad performances in the models come from misclassification and false positive prediction instead of failing to make a prediction that there is a polyp in the frame (misdetection). The exception being YOLOv3, which has high misdetection rate due to low confidence scores in its predictions. RetinaNet and FasterRCNN tend to make more false positive predictions on non-polyp frames than the other models.

From the video sequence analysis, we found that video sequences 2, 7, 8, 23, and 24 are particularly difficult for all the models, while video sequences 1, 13, 16, 20, and 21 are relatively easy for the models. RefineDet has the best performance on most of the hyperplastic sequences. Finally, the combined performance shows promising improvement potential results for this dataset.

In conclusion, RefineDet is the best performing model, having constant good performance in most of our analysis. This is also largely due to its good performance on the hyperplastic frames, which outperforms all the other models significantly. For adenomatous frames, however, SSD has the best performance and recall score.

Chapter 7

Future Works

Based on our comparison results, we will try to improve the performance of RefineDet, the best performing model from our analysis. The first improvement idea that we can test is to use DetNet as the backbone for this model. DetNet was designed specifically to replace commonly used backbones for an image classification task. The authors argued that commonly used image classification backbone lose more spatial information (which is needed for object detection task) due to the many pooling layers they use. The proposed DetNet architecture tries to solve this problem to improve object detection performance. Thus, we want to see if using DetNet as the backbone for RefineDet can improve its performance.

The second improvement idea that we can also test is the use of α weight factor as in RetinaNet focal loss. Since most of the prediction errors in RefineDet come from misclassification error, the use of a weighting factor might help the model to distinguish the two different classes. RefineDet architecture has a mechanism to deal with background-foreground imbalance problem using separate ARM and ODM modules. However, based on our analysis, we still need to improve its performance on distinguishing the two different classes. Distinguishing the two different classes in polyp detection and classification is difficult because sometimes the two different polyp classes may look similar from certain viewpoints. Thus, we still need to find a new idea to help improve the model's performance on classification.

References

Ballinger, A. B. & Anggiansah, C. (2007). Colorectal cancer. *BMJ*, 335(7622), 715–718.

Bechtold, M. L., Mir, F., Puli, S. R., & Nguyen, D. L. (2016). Optimizing bowel preparation for colonoscopy: a guide to enhance quality of visualization. *Annals of Gastroenterology: Quarterly Publication of the Hellenic Society of Gastroenterology*, 29(2), 137.

Bernal, J., Sánchez, F. J., Fernández-Esparrach, G., Gil, D., Rodríguez, C., & Vilariño, F. (2015). Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized Medical Imaging and Graphics*, 43, 99–111.

Bernal, J., Sánchez, J., & Vilariño, F. (2012). Towards automatic polyp detection with a polyp appearance model. *Pattern Recognition*, 45(9), 3166 – 3182. Best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2011).

Bernal, J., Tajkbaksh, N., Sánchez, F. J., Matuszewski, B. J., Chen, H., Yu, L., Angermann, Q., Romain, O., Rustad, B., Balasingham, I., Pogorelov, K., Choi, S., Debard, Q., Maier-Hein, L., Speidel, S., Stoyanov, D., Brandao, P., Córdova, H., Sánchez-Montes, C., Gurudu, S. R., Fernández-Esparrach, G., Dray, X., Liang, J., & Histace, A. (2017). Comparative validation of polyp detection methods in video colonoscopy: Results from the miccai 2015 endoscopic vision challenge. *IEEE Transactions on Medical Imaging*, 36(6), 1231–1249.

de Haan, M. C., van Gelder, R. E., Graser, A., Bipat, S., & Stoker, J. (2011). Diagnostic value of ct-colonography as compared to colonoscopy in an asymptomatic screening population: a meta-analysis. *European radiology*, 21(8), 1747–1763.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Eliakim, R., Yassin, K., Niv, Y., Metzger, Y., Lachter, J., Gal, E., Sapoznikov, B., Konikoff, F., Leichtmann, G., Fireman, Z., et al. (2009). Prospective multicenter performance evaluation of the second-generation colon capsule compared with colonoscopy. *Endoscopy*, 41(12), 1026–1031.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010a). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2), 303–338.

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010b). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2), 303–338.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440–1448).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

Group, U. C. S. W. (2018). U.s. cancer statistics data visualizations tool, based on november 2017 submission data (1999–2015): U.s. department of health and human services, centers for disease control and prevention and national cancer institute. www.cdc.gov/cancer/dataviz.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9), 1904–1916.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jass, J. R. (2004). Hyperplastic polyps and colorectal cancer: is there a link? *Clinical Gastroenterology and Hepatology*, 2(1), 1–8.

Johnson, C. D., Chen, M.-H., Toledano, A. Y., Heiken, J. P., Dachman, A., Kuo, M. D., Menias, C. O., Siewert, B., Cheema, J. I., Obregon, R. G., et al. (2008). Accuracy of ct colonography for detection of large adenomas and cancers. *New England Journal of Medicine*, 359(12), 1207–1217.

Johnson, C. D., MacCarty, R. L., Welch, T. J., Wilson, L. A., Harmsen, W. S., Ilstrup, D. M., & Ahlquist, D. A. (2004). Comparison of the relative sensitivity of ct colonography and double-contrast barium enema for screen detection of colorectal polyps. *Clinical gastroenterology and hepatology*, 2(4), 314–321.

Karkanis, S. A., Iakovidis, D. K., Maroulis, D. E., Karras, D. A., & Tzivras, M. (2003). Computer-aided tumor detection in endoscopic video using color wavelet features. *IEEE transactions on information technology in biomedicine*, 7(3), 141–152.

KIM, D. H. & PICKHARDT, P. J. (2010). Chapter 1 - colorectal polyps: Overview and classification. In P. J. Pickhardt & D. H. Kim (Eds.), *CT Colonography: Principles and Practice of Virtual Colonoscopy* (pp. 3 – 9). Philadelphia: W.B. Saunders.

Kim, N. H., Jung, Y. S., Jeong, W. S., Yang, H.-J., Park, S.-K., Choi, K., & Park, D. I. (2017). Miss rate of colorectal neoplastic polyps and risk factors for missed polyps in consecutive colonoscopies. *Intestinal research*, 15(3), 411.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

Lebwohl, B., Kastrinos, F., Glick, M., Rosenbaum, A. J., Wang, T., & Neugut, A. I. (2011). The impact of suboptimal bowel preparation on adenoma miss rates and the factors associated with early repeat colonoscopy. *Gastrointestinal endoscopy*, 73(6), 1207–1214.

Leufkens, A., Van Oijen, M., Vleggaar, F., & Siersema, P. (2012). Factors influencing the miss rate of polyps in a back-to-back colonoscopy study. *Endoscopy*, 44(05), 470–475.

Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., & Sun, J. (2018). Detnet: A backbone network for object detection. *arXiv preprint arXiv:1804.06215*.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980–2988).

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B., & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42, 60–88.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., & Pietikäinen, M. (2018). Deep learning for generic object detection: A survey. *arXiv preprint arXiv:1809.02165*.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21–37).: Springer.

Mesejo, P., Pizarro, D., Abergel, A., Rouquette, O., Beorchia, S., Poincloux, L., & Bartoli, A. (2016). Computer-aided classification of gastrointestinal lesions in regular colonoscopy. *IEEE Transactions on Medical Imaging*, 35(9), 2051–2063.

Mohammed, A., Yildirim, S., Farup, I., Pedersen, M., & Hovde, Ø. (2018). Y-net: A deep convolutional neural network for polyp detection. *arXiv preprint arXiv:1806.01907*.

Ngu, W. S. & Rees, C. (2018). Can technology increase adenoma detection rate? *Therapeutic advances in gastroenterology*, 11, 1756283X17746311.

Park, S., Lee, M., & Kwak, N. (2015). Polyp detection in colonoscopy videos using deeply-learned hierarchical features. *Seoul National University*.

Razzak, M. I., Naz, S., & Zaib, A. (2018). Deep learning for medical image processing: Overview, challenges and the future. In *Classification in BioApps* (pp. 323–350). Springer.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Redmon, J. & Farhadi, A. (2017). Yolo9000: Better, faster, stronger. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Redmon, J. & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Rees, C. J., Bevan, R., Zimmermann-Fraedrich, K., Rutter, M. D., Rex, D., Dekker, E., Ponchon, T., Bretthauer, M., Regula, J., Saunders, B., et al. (2016). Expert opinions and scientific evidence for colonoscopy key performance indicators. *Gut*, 65(12), 2045–2060.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (pp. 91–99).

Roland, C. L. & Barnett, C. C. (2009). Chapter 52 - colorectal polyps. In A. H. Harken & E. E. Moore (Eds.), *Abernathy's Surgical Secrets (Sixth Edition)* (pp. 258 – 261). Philadelphia: Mosby, sixth edition edition.

Saltzman, J. R., Cash, B. D., Pasha, S. F., Early, D. S., Muthusamy, V. R., Khashab, M. A., Chathadi, K. V., Fanelli, R. D., Chandrasekhara, V., Lightdale, J. R., et al. (2015). Bowel preparation before colonoscopy. *Gastrointestinal endoscopy*, 81(4), 781–794.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*.

Shinya, H. & Wolff, W. I. (1979). Morphology, anatomic distribution and cancer potential of colonic polyps. *Annals of surgery*, 190(6), 679.

Simon, K. (2016). Colorectal cancer development and advances in screening. *Clinical interventions in aging*, 11, 967.

Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*.

Society, A. C. (2019a). Cancer facts & figures 2019. *Atlanta: American Cancer Society*. https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2019/cancer-facts-and-figures-2019.pdf.

Society, A. C. (2019b). Cancer statistics center. http://cancerstatisticscenter.cancer.org.

Spada, C., Hassan, C., Munoz-Navas, M., Neuhaus, H., Deviere, J., Fockens, P., Coron, E., Gay, G., Toth, E., Riccioni, M. E., et al. (2011). Second-generation colon capsule endoscopy compared with colonoscopy. *Gastrointestinal endoscopy*, 74(3), 581–589.

Taha, B., Werghi, N., & Dias, J. (2017). Automatic polyp detection in endoscopy videos: A survey. In *2017 13th IASTED International Conference on Biomedical Engineering (BioMed)* (pp. 233–240).: IEEE.

Tajbakhsh, N., Gurudu, S. R., & Liang, J. (2015). Automatic polyp detection in colonoscopy videos using an ensemble of convolutional neural networks. In *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)* (pp. 79–83).: IEEE.

Van Rijn, J. C., Reitsma, J. B., Stoker, J., Bossuyt, P. M., Van Deventer, S. J., & Dekker, E. (2006). Polyp miss rate determined by tandem colonoscopy: a systematic review. *The American journal of gastroenterology*, 101(2), 343.

Wang, A., Banerjee, S., Barth, B. A., Bhat, Y. M., Chauhan, S., Gottlieb, K. T., Konda, V., Maple, J. T., Murad, F., Pfau, P. R., Pleskow, D. K., Siddiqui, U. D., Tokar, J. L., & Rodriguez, S. A. (2013). Wireless capsule endoscopy. *Gastrointestinal Endoscopy*, 78(6), 805–815.

Workshop, P. i. t. P. (2003). The paris endoscopic classification of superficial neoplastic lesions: esophagus, stomach, and colon: November 30 to december 1, 2002. *Gastrointestinal Endoscopy*, 58(6), S3–S43.

Zhang, S., Wen, L., Bian, X., Lei, Z., & Li, S. Z. (2018). Single-shot refinement neural network for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4203–4212).