21COP327 Artificial Intelligence Project

Prediction of Heart Disease using Machine Learning

Neville Abraham Mathew

September 2022

Contents

1 Abstract **1**

2 Acknowledgement **2**

3 Introduction **3**

3.1 Limitation & Gaps 4

3.2 Aims & Objectives 4

4 Literature Review **6**

4.1 Introduction 6

4.2 Heart Disease & its Types 6

4.3 State-of-the-Art ML Algorithms & Applications 8

4.3.1 Coronary Artery Disease ML Application 8

4.3.2 Congenital Heart Disease ML Applications 9

4.3.3 Summary & Discussion 11

5 Dataset and Experimental Setup **12**

5.1 Stenosis Detection Dataset 12

5.2 Data Preparation and Environment Setup 12

5.2.1 Data Loading Process 13

5.2.2 Dataset Split 13

5.2.3 Issues faced with Data Preparation 15

5.3 Summary 16

6 Algorithms & Methods **17**

6.1 Model Metrics 17

6.1.1 F1-score 17

6.1.2 Intersection Over Union Metric 17

6.1.3 MSE Loss Metric 18

6.2 Model Architecture Implementation 18

6.2.1 YOLO (You Only Look Once) 18

6.2.2 VGG-16 19

6.2.3 Sequential Model 21
6.3 Transfer Learning 21

7 Experimentation & Result Analysis 22
7.1 Result Analysis 22
7.2 YOLO Model 22
7.2.1 Fully Trained YOLO Visualizations 23
7.2.2 Transfer Learning YOLO Visualizations 23
7.3 Sequential & VGG-16 Model 26
7.3.1 VGG-16 Model 26
7.3.2 Sequential Model 26

8 Conclusion & Future Work 30
8.1 Hyperparameter Tuning 30
8.2 Functional API 31

List of Figures

5.1 ETL 13
5.2 High-Variance 14
5.3 Split-folders 15

6.1 YOLO Architecture 19
6.2 VGG-16 Architecture 20

7.1 Pretrained YOLO Results 24
7.2 Confusion Matrix 1 24
7.3 Transfer learning YOLO Results 25
7.4 Confusion Matrix 2 25
7.5 VGG Result 1 27
7.6 VGG Result 2 27
7.7 VGG Result 3 28
7.8 Seq Result 1 28
7.9 Seq Result 2 29

1 Abstract

This study examines the role of machine learning in making accurate predictions in the field of heart disease. In the 21^{st} century, current and ongoing cases of heart failure and diseases have increased tremendously, and solutions to prevent such cases are an ongoing challenge for physicians and healthcare researchers.

The topic chosen for this study will be the prediction of coronary artery stenosis (CAS). CAS is commonly referred to as heart attacks. Incorporating deep learning techniques can help predict the detection of stenosis in patient angiographic scans, thus reducing the strain on vascular surgeons. The original approaches used in the CAS paper included the use of pre-trained models such as ResNet, MobileNet, VGG etc. implemented through Amazon SageMaker. However, the results and implementation of the original approach will be used to compare with the custom deep learning models created using Python and its libraries, as shown in this study.

Furthermore, further research and development on custom algorithms will also be discussed to analyse how they can be implemented in other fields of disease detection, such as brain magnetic resonance imaging, electrocardiogram signal classification, etc. Closing toward the conclusion of the deep learning methods and deciding whether the results have been met.

2 Acknowledgement

I would like to take this opportunity to thank my supervisor, Dr. Meng who has provided me the guidance to start and complete this project. I am grateful towards the computer science department for providing me two weeks of extension to complete my work. I would also like to thank the IT support staff for helping me solve any underlying hardware or software issues at hand. Finally, I would like to thank my friends and familty who have helped me to solve issues and provided motivation throughout the project.

3 Introduction

The background of the project topic, as well as the objectives and goals, should be described in the introductory paragraph. The leading cause of death worldwide is cardiovascular disease (CVD), particularly heart attacks and heart failure. As the leading cause of death for women worldwide, CVD has claimed more lives than HIV/AIDS, malaria, and both together. Around 17.9 million people die from CVD each year, according to (Organization, 2019).

The use of deep learning can prevent further CVD deaths. To determine whether such a patient with chest pain is likely to suffer a heart attack or stroke, for example, an algorithm designed for a computer vision task can help with prediction and prevention. This can help medical professionals not to spend too much time labelling the data and manually detecting stenosis, which is a tedious task. These algorithms can help identify patients with a higher chance of heart rate in the future. They can be used to provide more accurate and timely predictions on the likelihood of advanced heart disease or stroke. Review of specific articles and their methods will improve the understanding of the algorithms that are being applied. The use of Convolutional Neural Networks (CNNs) can help solve the underlying problem of stenosis detection and data labelling.

CNNs have proven to be a competent tool for helping with image classification and computer vision tasks. Due to its high speed and precision of results, it has shown great promise in the detection of coronary artery disease (CAD). However, the biggest problem faced with CNN application can be called the speed/accuracy trade-off, where a model can produce accurate results but slow to produce the final output and vice versa Yang et al. (2019). For CNN models to be applied in real-time detection, it must be applicable to low-end healthcare equipment and mobile devices that can be used by physicians. Therefore, it is crucial to find the right balance between speed and accuracy for use in medical devices.

This study will introduce the application of Python deep learning models to help predict coronary artery stenosis. The work presented will highlight the work of custom models along with a pre-trained model for result comparison. The models include the YOLO (You Only Look Once, fully pre-trained network) model, the

VGG-16 model (using pre-trained weights), and transfer learning techniques. The pseudocode of the model and the results will be highlighted and visualised.

3.1 Limitation & Gaps

Although, deep learning algorithms have shown great results, for working with heart data, there is a gap that needs to be filled. This gap is described in Venkat (2020) and points out problems associated with data trust, insufficient data, problem syndrome, and lack of domain knowledge. These problems are critical and will prove why deep learning algorithms cannot replace medical specialists.

Data trust is a common culprit for false or inaccurate results when working with heart data from patients. Many times, data is not being processed or no quality checks are performed to ensure outliers, corrupted data or empty values do not exist. Sometimes, even if the data is processed correctly, the features are not sufficient to provide a valid prediction.

Insufficient data is one of the most common problems faced in providing prediction results to patients. To use deep learning techniques, it would require a lot of patient data to extract the necessary features. Deep learning models like ResNet, MobileNet etc. require a large amount of data, which can be problematic for hospitals to acquire patient data considering the privacy and safety measures involved.

The problem syndrome refers to entering the prediction task without performing exploratory data analysis (EDA), and this results in diving into the problem without defining what the actual problem is. Many instances may not require machine learning to solve a prediction problem. It is important to ask the ”why”, ”what”, and ”how” questions before going on to the issue.

Finally, Venkat (2020) states an important point, which is a lack of knowledge about cardiovascular diseases and a lack of trust in the algorithm to solve the required task. Regardless of the algorithm or the best metrics used for prediction, it is important to understand patient data and what features to use in model training. Thus, physicians can assume the role of providing domain knowledge to machine learning engineers.

3.2 Aims & Objectives

The aim is to discover how machine learning algorithms can help detect coronary artery disease in exposed patients. In addition to emphasising the value of cardiovascular image analysis through AI technologies, Seetharam et al. (2021) also discusses recent applications of artificial intelligence, such as Narula et al. (2016) and Zhang et al. (2018) use algorithms for supervised learning and deep learning.

The objective of the project is to provide a better method than the accepted one by using deep learning models. Determine whether the results are diagnostically valuable and should be communicated to patients with heart disease. The goal and objectives will be appropriately met through thorough research of related works.

4 Literature Review

4.1 Introduction

This section will provide an essential understanding of the field of heart disease with references to existing methodologies. The review of the literature will also highlight any progress in developing ideas or methods in the project. Machine learning and different technical aspects will be emphasised in relation to the related work done.

4.2 Heart Disease & its Types

Heart disease is related to different types of heart disease, each having its own symptoms and effects on the human body. According to the CDC, the United States is the most common type of heart disease, coronary artery disease (CDC, 2021). The common name for this is heart attack. The following two subsections will describe the two common types of heart disease.

Coronary Artery Disease (CAD)

CAD has been elucidated as a disease that occurs in both old and young people. According to CDC (2021), CAD occurs due to a blockage of the coronary arteries (blood vessels leading to the heart), resulting in a decrease in the flow of blood to the heart muscles, and therefore decreases the supply of oxygen to the heart. The early stages of CAD are called atherosclerosis, which is the start of the disease before its impact on the heart. Atherosclerosis is defined as the hardening or shrinkage of the coronary arteries caused by plaque build-up in blood vessels over time (Felson, 2021). The most common symptoms include chest pain, shortness of breath, indigestion, and many more (Donovan, 2018).

A study by Duda-Pyszny et al. (2018) shows the effect of CAD on women. The purpose of the study is to show the vital differences between the clinical characteristics of men and women along with the treatment of CAD in men and women. Before in-depth analysis is performed, it is important to understand what angiography X-rays show, as this is considered for the data in the study.

According to NHS (2017), an angiography is a type of radiograph that can help the doctor see the blood vessels of the heart. Angiograms are the X-ray images that are produced as a result of angiography. The clinical characteristics considered within the study include CAD symptoms, risk factors, non-invasive diagnostics of CAD, management of cardiovascular risk, and elucidation of angiograms for women.

CAD symptoms are unique for men and women in diagnosing angina pectoris (chest pain due to reduced blood flow). For women, common symptoms include sweating, nausea, fatigue, dyspnea, and many more, while for men, the same symptoms are less severe. Risk factors between both sexes are common, including smoking, age difference, and genetic background. Non-invasive diagnostics of CAD involve understanding whether the patient is at risk of acquiring CAD, where women show a higher resting heart compared to men on the same tests. The risk of CAD treatment is the same for both men and women, while patients with high cardiovascular risk generally show angina pectoris. Lastly, angiography still appears to be the best method of analysing coronary artery stenosis angiograms in men and women.

Finally, Duda-Pyszny et al. (2018) show that CAD treatment reduces symptoms and improves recovery, however, both men and women were shown to be less likely to receive proper treatment compared to men.

Congenital Heart Disease (CHD)

According to Beckerman (2002), congenital heart disease refers to a defect in heart formation while a baby is still in the womb, resulting in multiple heart problems once the baby is born. Although some symptoms can occur immediately after birth, directly into adulthood. According to CDC (1998), CHD is the most common birth defect that affects approximately 40,000 births in the United States each year; some of the defects also lead to infant deaths. Septal abnormality or atrial septal defect is a common CHD problem in which the wall that separates the left and right sides of the heart has holes, and surgery would be required to correct this problem (Mai et al., 2019). Common symptoms include lung infections, swelling in certain parts of the body, rapid heartbeat, fatigue, and others (DeNoon, 2007).

A study by Strah et al. (2021) aims to show the differences between patients (children and adults) who are COVID-19 positive for coronary heart disease compared to patients without coronary heart disease. Data are acquired from the Vizient clinical database consisting of patient admissions for COVID-19 and mild to severe coronary heart disease from April 2020 to March 2021. Admission data present 9,478 children with COVID-19 with 160 children who are positive for coronary heart disease and 658,230 adults with COVID-19 and 389 adults with coronary heart disease. Strah et al. (2021) have provided information on the comparison between adults and children in a few factors such as; admissions rate, hospital stay period, complications, mortality rate, and costs.

For children diagnosed with COVID-19 and coronary heart disease, the admission rate was higher in the younger age groups of 1 to 11 years, resulted in longer hospital stays (22, 6 days), more complications (6.9, 1.1%), a higher mortality rate (3.8, 0.8%) and higher costs (54,619,$10,731). The same can be shown for adults diagnosed with COVID-19 and coronary heart disease, the results indicated that younger age groups such as 53 compared to 64 years showed more admissions, a longer hospital stay (12, 9 days), a higher complication rate (8, 4.8%) and higher costs (23,551, $13,311). Therefore, Strah et al. (2021) has concluded the study, stating that patients diagnosed with COVID-19 and coronary heart disease pose a greater risk/threat to their overall health, where children of younger ages have higher mortality rates.

4.3 State-of-the-Art ML Algorithms & Applications

This section will include recent studies on the applications of machine learning algorithms for CAD and CHD, respectively.

4.3.1 Coronary Artery Disease ML Application

The following studies show the application of neural networks and traditional machine learning for the detection of CAD.

Coronary Artery Stenosis Detection with Neural Networks

This specific heart disease will be the main focus of the final MSc project. A common type of CAD called coronary artery stenosis proposed by Danilov et al. (2021a) shows the use of neural networks to help detect stenosis efficiently and effectively. The objective of the study is to help determine the feasibility of real-time detection of stenosis through deep learning. Danilov et al. (2021a) have tested different neural network architectures, which are ResNet-50 and 101, Inception ResNet, MobileNet, and NASNet. Models were implemented on 100 patient angiography scans. Of the five models tested, Danilov et al. (2021a) have shown that only three models were pronounced superior in the detection of stenosis. The Inception ResNet model scored a mean average precision score of 0.95, an f1 score of 0.96, and a mean prediction rate of 3 fps (frames per second). The second model, MobileNet, achieved a mean average precision score of 0.83, an f1 score of 0.80, and a mean prediction rate of 38 fps. Finally, the third model, ResNet-101, achieved a mean average precision score of 0.94, an f1 score of 0.96, and a mean prediction rate of 10 fps. Therefore, the

Inception model is the most accurate, MobileNet is the fastest, and the ResNet-101 model is the best overall model with a good balance of accuracy and prediction rates.

Coronary Artery Disease Detection using Machine Learning with Coronary Bifurcation Features

Another application of the CAD ML application is shown in Chen et al. (2020), which highlights various ML models that are being tested for CAD detection. The key difference from this study compared to the above research paper is that there is a feature extraction method. The features are extracted by Chen et al. (2020) using the morphometric method (Pereira et al., 2010). The selected characteristics are called coronary bifurcation characteristics, which consists of the selection of certain coronary vessels within the heart, such as the left and right vessels (Medrano-Gracia et al., 2016).

The data used in the article were acquired from coronary computed tomography angiography (CCTA) scans. According to Chen et al. (2020), the results were based on the trial of six machine learning classifiers, which are: Decision Tree (DT), Artificial Neural Networks (ANN), Support Vector Machines (SVM), K-Nearest Neighbours (k-NN), Logistic Regression (LR), and Linear Discriminant Analysis (LDA).

The results were as follows on the basis of the test data: DT for 97%, ANN for 98.4%, SVM (Polynomial-SVM) for 100%, k-NN for 95.7%, LR for 96.3% and LDA for 92.3%. Although the six models would provide perfect predictions and speed, the SVM model demonstrated a complete 100% accuracy for the predictions, and the LDA classifier proved the least accurate.

4.3.2 Congenital Heart Disease ML Applications

The following studies show the application of deep neural networks and traditional machine learning to detect coronary heart disease.

Heart and Vessel Segmentation of CHD using Deep Neural Networks and Graph Matching

The paper provided by Xu et al. (2019) shows evidence of the impact of coronary heart disease in children as the leading cause of high birth mortality rates in the United States. Xu et al. (2019) aim to create a new framework in which deep learning can help solve the segmentation of the whole heart and vessels for coronary heart disease cases. CHD segmentation is not considered a normal type of segmentation, as CHD hearts have unique variations. The data used in the article is 68 CT scans that include 14 different types of CHD, which are Atrio-Ventricular Septal

Defect (AVSD), Co-Arctation (CA), Pulmonary Stenosis (PS), Atrial Septal Defect (ASD), Patent Ductus Arteriosus (PDA), Ventricular Septal Defect (VSD), Aortic Arch Anomalies (AAA), Tetrology of Fallot (ToF), Transposition of Great Arteries (TGA), Common Arterial Trunk (CAT), Pulmonary Artery Sling (PAS), Anomalous Drainage (AD), Single Ventricle (SV) and Pulmonary Atresia (PuA).

The framework provided in the paper shows a series of techniques, which are region-of-interest cropping, chambers and myocardium segmentation, and chambers and myocardium refinement. This process will be the entire segmentation of heart scans for coronary heart disease. After the heart segmentation is complete, the vessel segmentation will involve two other techniques, such as blood pool segmentation and graph matching.

The experimentation was performed on two models, which are 3D U-Net and Seg-CNN (original method for comparison). The results showed that the work of Xu et al. (2019) was shown to have a higher mean and standard deviation result compared to the original state-of-the-art method, 3D U-Net (mean = 82.6 and 74.1%, standard deviation. = 6.2 and 14.5%) providing improved precision over Seg-CNN (mean = 70.3 and 62.7%, standard deviation = 8.3 and 14.4%) for milder cases of coronary heart disease and cases of severe coronary heart disease, respectively.

Machine Learning Methods to Identify Predictors for CHD

An article by Qu et al. (2022) elaborates on ML methods to help predict the early onset of coronary heart disease before birth, thus reducing the mortality rate. The objective of the study is to understand different risk factors, which can also be external factors, and to consider indicators to predict whether the baby will have coronary heart disease or not.

Data used for the study were obtained through a cardiac centre in China recorded between 2011 and 2017, and the respective CT scans were verified by the responsible paediatricians. The extraction and selection of features is included in this paper. Through feature extraction, 1,127 predictors were extracted to serve as indicators of coronary heart disease. However, the process of feature selection considers the top predictors with the highest threshold values.

An Explainable Boosting Machine (EBM) was considered the best choice for the machine learning prediction model (Nori et al., 2019). The final results of this ML model scored results of accuracy, specificity, AUC, and sensitivity parameters which are 65%, 65%, 76% and 74% respectively.

4.3.3 Summary & Discussion

The algorithmic application to the detection of heart disease in different types of disease is unique and complex. Working with different types of patient data requires specialists to label and filter out outliers. CAD and CHD detection shows great promise in terms of how neural network applications can improve detection, as shown by the above-mentioned results. However, the issue faced with researchers is that the implementation of neural nets such as ResNet, 3D U-Net, MobileNet etc. is related to portability. Working with hospital technology can be outdated and unpredictable for implementing neural networks, as these models require high-end technology if the goal is to get better results compared to just the speed of the models itself. This question poses a great challenge to whether deep learning can be fully optimised with these challenges (Seneviratne et al., 2020). The challenges of these applications do not end here; researchers have tried to bridge the gap between healthcare and AI, including algorithm bias, costs, ethical requirements for data usage, etc.(Kelly et al., 2019). Continuous improvement of such models should not be the only factor for using deep learning models in detection; having a clear adoption strategy and training amongst physicians is important.

To wrap up this discussion, AI technology in healthcare is facing criticisms and challenges. Researchers must understand how to deal with these problems and find appropriate solutions to them. No matter how accurate the deep learning model is in detecting a disease, physicians must always make the final decision for the patient. Forecasting the future of AI in healthcare shows promise, with well-founded criticism.

[1]

[1] Literature Review taken from Neville's Project Preparation CW (COP324)

5 Dataset and Experimental Setup

5.1 Stenosis Detection Dataset

This project involves the use of a medical dataset which is based on the detection of coronary stenosis. The data involves the use of angiographic images with labeled stenosis cases in each of the scans. The scans were captured at a research institute for cardiovascular diseases located in Russia using medical through patients undergoing coronary angiography. Image-guided surgery equipment such as Coroskop and Innova were used for image capturing. The Siemens Coroskop machine is a catheter lab specifically created for minimally invasive procedures to acquire important features of the heart patients. A catheter lab such as Coroskop can provide many surgeries such as angiogram, ablation and angiography which is handled by a cardiologist (BHF, 2020). The Innova machine by GE healthcare is another type of minimally invasive procedure where it provides access to procedures for cardiovascular, oncology, electrophysiology etc. This machine is far more advanced than Coroskop where the main difference is in 3D imaging and video (Healthcare, 2020).

The patients were all confirmed with one case of coronary artery disease which is shown in the angiography scans, the dataset only contains positive cases of stenosis. Additionally, the patients provided a written consent for participation and data use in the study. Furthermore, the dataset was gone through heavy filtering by a cardiologist to remove any non-informative images and only use the well-defined stenosis cases. The dataset contains 8325 grayscale images of 100 patients in sizes of 512x512 - 1000x1000 which are labeled using LabelBox, an AI software used to label unstructured data. The labeling process was performed by a specialized cardiologist (Danilov et al., 2021b).

5.2 Data Preparation and Environment Setup

Although the images came readily labeled and processed, a lot of data preparation was required to split the dataset, resize the images, match labels according to the image names, install required environment dependencies, model preparation etc.

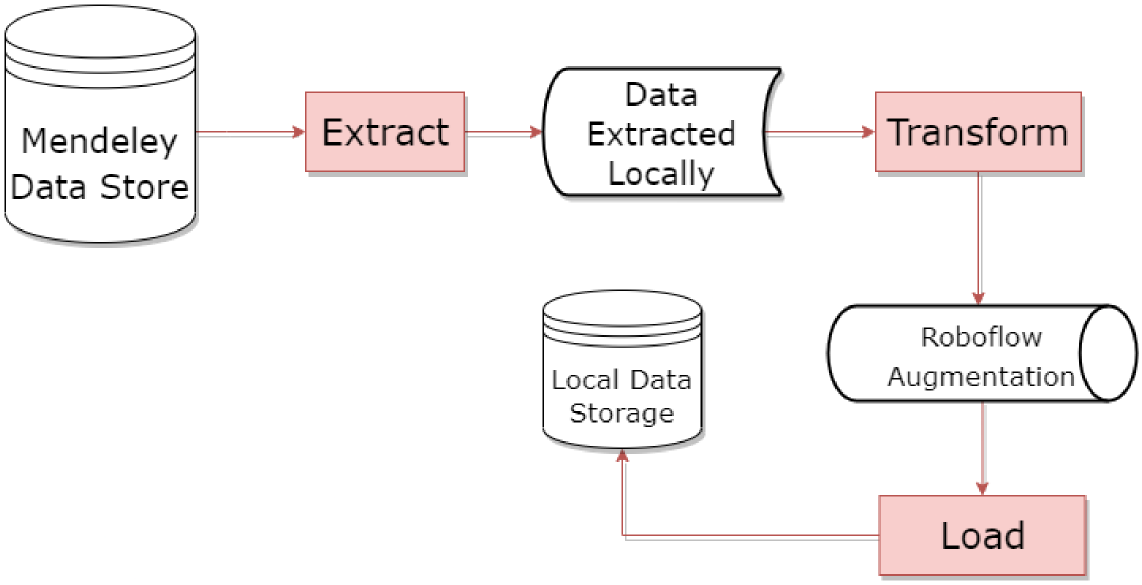

Figure 5.1: ETL

5.2.1 Data Loading Process

After the images were split into their respective folders, the labels and image resizing were worked on. All images were resized from sizes such as 512x512, 600x600, 1000x1000 etc. to 224x224. The reason for this image dimension is to help for model training speed and performance, running on higher resolution images will lead to out-of-memory errors and high training time for each epoch stage. After the images were resized, the labels were then scaled for inputting to the model.

An open source software named Roboflow was used for the purpose of image resizing and label adjustment. Roboflow provides the privilege of loading bulk data with annotations such as the stenosis dataset which includes high resolution images. Roboflow solves the issue of memory overload which is common for large data loading procedures and performs ETL (Extract, Transform and Load) functions (5.1). The data loading process involves getting authorization for data access, extracting the data into the local machine, transform the data using Roboflow for image resizing and labeling the images according to the new dimensions, finally the processed data is loaded back to the local machine. The image below illustrates the data loading process.

5.2.2 Dataset Split

According to Danilov et al. (2021a), the data was reportedly split with 80-10-10 split which is 5993 images for training, 833 images for validation and 832 images for testing. The current study will follow a different split of 70-20-10 which is 5827 images for training, 1665 images for validation and 833 images for testing. This will help solve the issue of high variance, having too small of a validation set will not

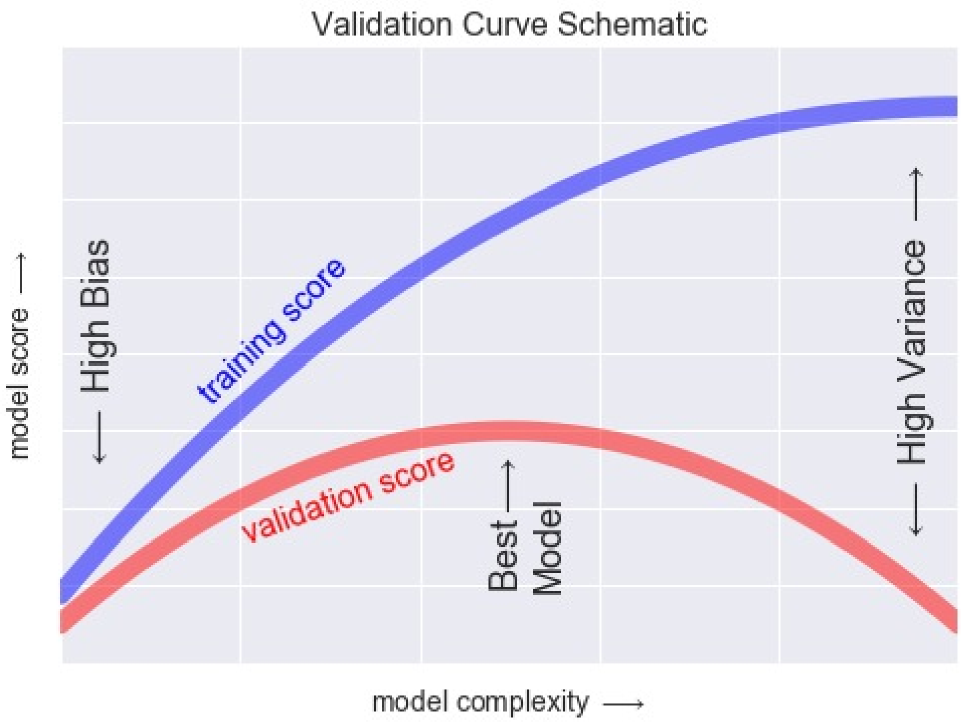

Figure 5.2: High-Variance

show good results for real time prediction on the test set. Apart from the images, the labels were also preprocessed which will be discussed in-depth further down. The figure (7.2) shows a basic understanding of high bias model and high variance which leads to underfitting and overfitting respectively.

Additionally, a small pseudo-code snippet will provide knowledge on the process of the dataset split (??). A library called 'split-folders' will be used to help easily split the data into an 80-20 split for train and validation. For the testing set, a simple if-else conditional loop will be used to help further split the respective labels. For test set images, the validation set will be further divided according to the the test labels CSV file.

Data: stenosis images
Result: how to split validation for test set
initialization;
for *filename in test labels csv file* **do**
  copy respective images from the label filenames to new folder;
end

Algorithm 1: Using shutil to split images for test set

IMPORT libraries including splitfolders and pandas

IMPORT dataset and labels

READ dataset and labels

STORE dataset and labels **INTO** variables

INITIALIZE splitfolders with train-test split

APPEND to list variable

Figure 5.3: Split-folders

Data: stenosis labels
Result: how to split labels for test set
initialization;
while *complete labels file* **do**
  read csv file using csv reader;
  while *open test csv file* **do**
    write to new csv file using csv writer;
    for *read all rows in csv file* **do**
      if *filename exists in labels file and write to new test csv file* **then**
        write to new csv file if condition is True;
      end
    end
  end
end

Algorithm 2: Test labels split

5.2.3 Issues faced with Data Preparation

1. Handling large number of high resolution images led to many complications with memory issues and model training. Resizing the images and converting the labels and images to numpy arrays increased speed and prevented kernel crashes.

2. Setting up the anaconda environment was a huge complication as it led to issues with using the GPU for running the code scripts. The issue was solved using a specific Python version of 3.9.0 with Tensorflow version of 2.8.1.

3. Due to the images being of a different size, the labels had to be normalized

according to the image target size. If the labels were not normalized, this would lead to negative values in our model. This problem was solved using

5.3 Summary

To summarize the dataset preparation, more augmentations could have been performed to the images such as blurring, flipping, cropping etc. However, for the stenosis images there would be no reason for any additional augmentation as the stenosis features would not be accurate for the model input. After the dataset is prepared, the features and labels are then inputted to the model for object detection which will be further explained in the next section.

6 Algorithms & Methods

This chapter will introduce the models and the concept of transfer learning used for this study, along with the metrics used for model evaluation and prediction. The models are YOLO, VGG16 and other custom models made using Tensorflow's keras library in Python. The metrics used for evaluating our models will be IOU (Intersection Over Union), F1-score and loss.

6.1 Model Metrics

A concise summary of the metrics used with their mathematical annotation will provide a clear understanding of the model evaluation process.

6.1.1 F1-score

The f1-score, also called F-measure is an error metric which is used for measuring model performance through calculating the precision and recall values. This metric can help understand how accurate the predictions are if a dataset is imbalanced or balanced. However, interpreting the values is important for model performance capability. The f1-score will have values ranging from the lowest of 0 to the highest of 1 which means the best predictions. The values are interpreted through capturing the positive cases of recall and precision. Having a score greater than 0.9 is considered to be very good while having a score less than 0.5 is not good (Allwright, 2022).

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{6.1}$$

6.1.2 Intersection Over Union Metric

The IOU metric shows how well the predicted bounding boxes are closely fit to the target bounding box coordinates. The intersection over union is calculated by taking the ratio of the intersection area with the union of the predicted and target/ground truth bounding boxes. IOU will decide if the bounding box is a true positive, false positive or false negative. True negative will not be considered since the assumption of the object is present in each image. True positive will mean there is an IOU higher than 0.5, this shows a correct or close enough bounding box prediction. A false positive will mean an IOU value lower than 0.5 or having duplicate bounding box predictions i.e. two predictions in the same image. Finally, false negative will mean no prediction presented at all (Tan, 2022).

$$IOU = \frac{Area\ of\ Intersection\ of\ two\ boxes}{Area\ of\ Union\ of\ two\ boxes} \tag{6.2}$$

6.1.3 MSE Loss Metric

The MSE or mean square is the mean square difference between predictions and ground truth values. This loss function provides an easy way to calculate gradients, however, if predictions are too high than the ground truth values it will lead to penalization over the loss values. The lower the mse loss value would lead to better prediction rate (Parmar, 2018).

$$MSE = \frac{\sum_{i=1}^{x} (y_i - \hat{y})^2}{n} \tag{6.3}$$

6.2 Model Architecture Implementation

6.2.1 YOLO (You Only Look Once)

The YOLOv5 model (6.1) is a pretrained model created by Ultralytics in June 2020 which is heavily used for object detection tasks. It provides high accuracy of real-time object detection within images and video data. The model implements the use of a single neural network which processes the image into separate parts and predicts the bounding box coordinates with probabilities. The model architecture is made up of three components which is; Backbone, Neck and Head.

The backbone consists of feature extraction from the input image, CSP (Cross Stage Partial Networks) is used as the backbone for extracting the features (Garg, 2021). Cross stage partial network is a type of convolutional neural network which is used as a common backbone for object detection tasks (Bochkovskiy et al., 2020). The CSP network is heavily used with Darknet, which is an open source neural framework created with C and CUDA. The CSPNet has proven good results for object detection tasks in models such as CSPResNet-50, CSPDarkNet-53, CSPResNeXt-50 and more (Wang et al., 2020). The Neck consists of feature pyramids which help in the object detection task and working with unseen data. PANet will be used in the Neck of the YOLO model to generate feature pyramids (Garg, 2021). PANet (Path Aggregation Network) is responsible for the information flow for the instance seg-

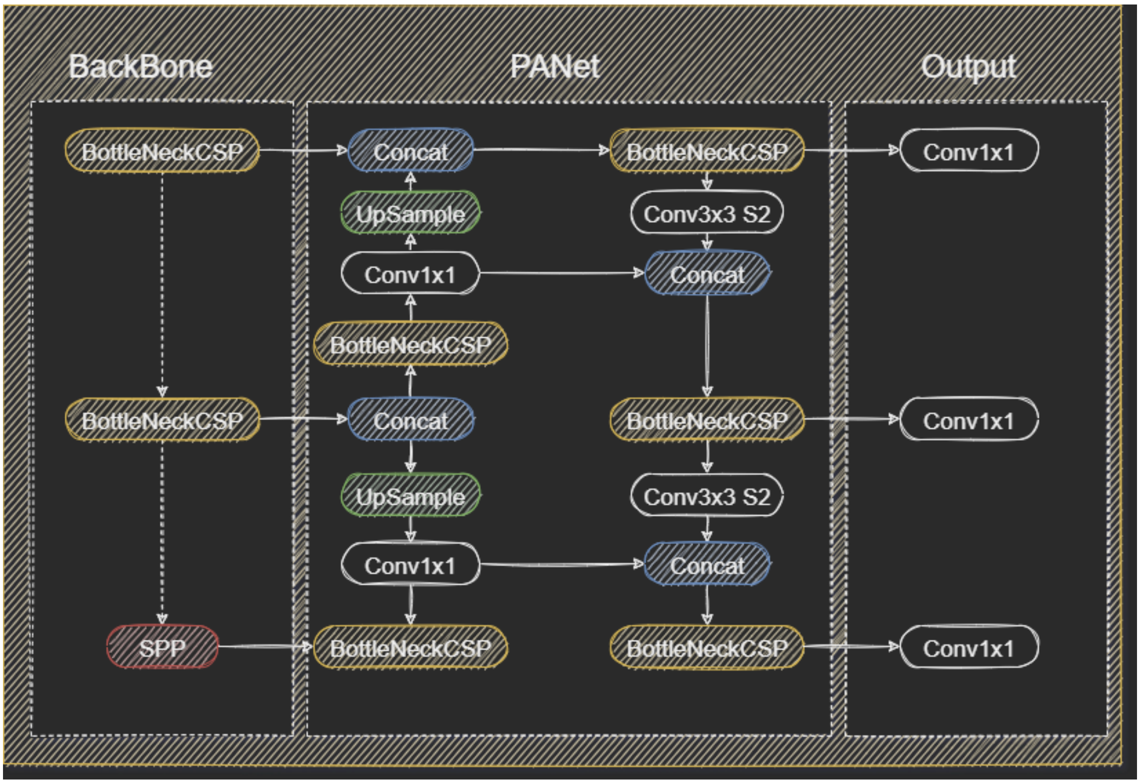

Figure 6.1: YOLO Architecture

mentation framework which shortens the path between the backbone and head parts of the YOLO network. The PANet framework was invented to improve information propagation by following a bottom-up path augmentation method which in turn improves feature extraction through adaptive pooling which combines feature grids with the feature levels to project information (Liu et al., 2018). Lastly, the head will be the last part of the YOLO network which is responsible for the detection and construction of the bounding box predictions (Garg, 2021).

For the current study, the YOLO model will be trained as a full model and separately trained with transfer learning. The fully trained YOLO network is pretrained on the COCO dataset which will provide good results for object detection tasks. However, this will not be indicative of real-time performance, therefore, transfer learning will be done to help understand the real-time performance of the model on the stenosis dataset. For the transfer learning technique, the backbone will be frozen which consists of 10 layers. These 10 layers are responsible for the feature extraction.

6.2.2 VGG-16

The VGG (Visual Geometry Group) (6.2) model is a convolutional deep neural network which was created through the work of Oxford researchers which is comprised of convolutional blocks with max pooling layers. There are two types of VGG models with 16 and 19 layers deep. For the project, VGG-16 will be used as the model

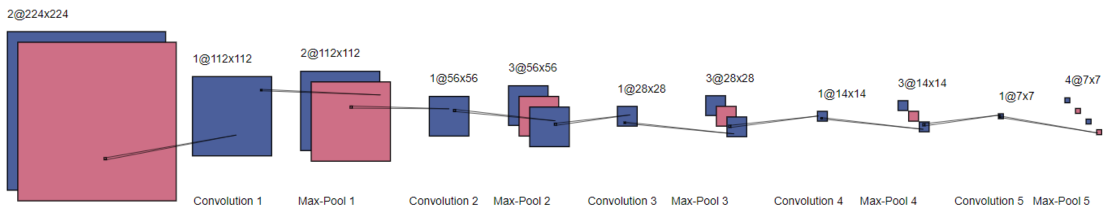

Figure 6.2: VGG-16 Architecture

performance is just as good with better speed since lesser convolution blocks. The VGG model is trained on the imageNet dataset challenge which is to classify over 14 million images with their respective labels. The model received 92.7% accuracy with images belonging to 1000 classes. The architecture of VGG-16 takes input of 224x224x3 images.

The first two convolution layers have 64 channels with filter size and padding of 3x3, after those two layers a max pooling layer of (2,2) stride is added. The next convolutional block follow the same structure as the previous two layers with max pooling, however there are 128 filters for the convolution layers. The convolutional block structure following after is three convolution layers with one max pooling layer in each block. The number of filters increases towards the end with 256, 256 and 512. With each image passing through the convolution and max pooling layers, the input size of the image becomes smaller through each max pooling. The output will result in a feature map which will be flattened into a feature vector through which it will pass through to two fully connected layers which output a (1, 4096) vector for each FC layer. The last fully connected layer will be the output layer of 4 classes which is the bounding box values for the object detection data. The default activation function for the fully connected layers is 'ReLU' (Rectified Linear Unit).

The problem faced while working with the VGG16 model is the slow training time and the problem of exploding gradient. The exploding gradient problem refers to large error gradients promote huge update to the network weights during model training. This can result in spikes of loss values, poor model prediction or even no prediction at all (DeepAI, 2019).

The project will include the use of VGG16 as a model with transfer learning. The transfer learning technique will freeze all the classification layers of the model and only use the imagenet weights to provide bounding box predictions. The results of this model will be compared with the predictions of the YOLO model to assess which provides better quality predictions using the above highlighted metrics. In addition, the original approaches of Danilov et al. (2021a) will also be considered in the model comparison.

6.2.3 Sequential Model

The sequential model is a custom model made through Keras library which allows the user to add as many feature layers and fully connected layers as seen appropriate. This allows the user to create a model according to the use case. Most pre-trained models have large number of parameters and are too slow for model training. With a sequential model, hyperparameter tuning or even manual tuning can be performed to find the best model fit for the current use case. The sequential model type proposes a linear stack of layers with input and output as only one tensor.

The current sequential model will have 13 convolutional blocks with leakyrelu activation layers, the number of filters are increasing from 32 down to 512 for the feature extraction process. After the convolution layers, global average pooling layer is then added which changes the shape of the input tensor to a batch size and channels last format before moving to the fully connected layers. The full connected layers include three Dense layers with ridge regression on for the first two layers, the neurons are decreasing from 2048 to 1024 and finally the last Dense layer will be the output which is 4 labels, followed by a sigmoid activation.

6.3 Transfer Learning

For computer vision tasks, neural networks are able to detect the edges of input images from the first layer to the final feature extraction layer. Transfer learning can help improve prediction results on custom dataset with the help of a pretrained model utilizing imageNet weights where the pretrained model will have the convolution layers frozen or untrainable and use only the weights with the Dense layers. Although, transfer learning is performed on smaller datasets to increase performance, the stenosis dataset is large which should improve prediction rate with the VGG-16 and YOLO models (Sharma, 2021).

For the study, VGG-16 and YOLOv5 model will show performance with and without the transfer learning technique. It is crucial to highlight the differences in the results between the models and how each perform with various data samples.

7 Experimentation & Result Analysis

The final objective of the project is analyze and study how stenosis detection can be performed through the work of deep learning techniques. In this chapter, the results of the work performed will be shown and compared to the original approaches.

7.1 Result Analysis

For this section, few experiments will be shown with YOLO, VGG-16 and the custom model. The pretrained models will be using transfer learning to help with model performance and the custom sequential model will be trained from scratch.

7.2 YOLO Model

The table above showcases the metric comparison of few of the original approaches from Danilov et al. (2021a) and the YOLO model implementation. As shown, the fully pretrained YOLO model outperforms the other models in F1 and mAP scores. However, transfer learning with the YOLO model does not perform as well as the other pretrained models. This could be due to lower training epochs or variation in learning rate. This proves a huge scope for improvement of using YOLO with

Model	Metrics	
YOLOv5 (Full pretrained)	mAP	0.918
	F1	0.91
YOLOv5 (Transfer Learning)	mAP	0.735
	F1	0.73
SSD MobileNet V1	mAP	0.69
	F1	0.72
SSD MobileNet V2	mAP	0.83
	F1	0.8
SSD ResNet-50 V1	mAP	0.76
	F1	0.73

freezing layers by transfer learning.

The YOLO model is trained on the COCO dataset which is far more comprehensive than the imageNet dataset which was trained by the MobileNet models which could be the reason for lower performance. Furthermore, single shot detector (SSD) models utilizes convolutional blocks which then go through the input images at higher precision and accuracy as a balance whereas YOLO learns the object detection task through the required bounding box corrdinates which provide a higher prediction rate at the cost of speed.

7.2.1 Fully Trained YOLO Visualizations

The figure (7.1) shows the visualization result of the YOLO fully pretrained model which shows the prediction scores for stenosis on 16 images which have about 0.8-0.9% IOU prediction rate, this is a high prediction value for each of the images. Only one of the 16 images has a prediction rate of 0.6%. This performance is to be expected as all layers are present and pretrained weights are present. The f1 score also has superiority of 0.91 amongst all classes. Additionally, metrics such as precision, recall and mAP have also shown promising results which are 0.891, 0.96 and 0.918% respectively. Using these metrics, it can be proven that the fully trained YOLO model works well for the detection of stenosis.

The confusion matrix (7.2) shows the classification of true positives, true negatives, false positive and false negatives. The false positive value is 1.00 which means that stenosis was not predicted at the right place in a few images, known as type 1 error. However, the true positive value is 0.93 which means the object was positive and predicted correctly. True negative is shown as blank since it is confirmed stenosis is present in each image. Finally, the false negative value is low at 0.07 which means that stenosis present in a few images were predicted as negative, known as type 2 error.

7.2.2 Transfer Learning YOLO Visualizations

The figure (7.3) highlights the work of transfer learning on the stenosis dataset, the results were not as promising as compared to the fully trained YOLO model. From the 16 images, it can be see most of the IOU prediction rate is around 0.4-0.6% which is not considered as an accurate prediction. Additionally, the predictions across the images are showing two times on one image which is false since each image is proposed to have one class of stenosis. This type of prediction results would not be considered for real-time medical performance. Although, the model could be improved by training for longer time using a lower learning rate, adding ridge or lasso regression techniques and hyperparameter tuning.

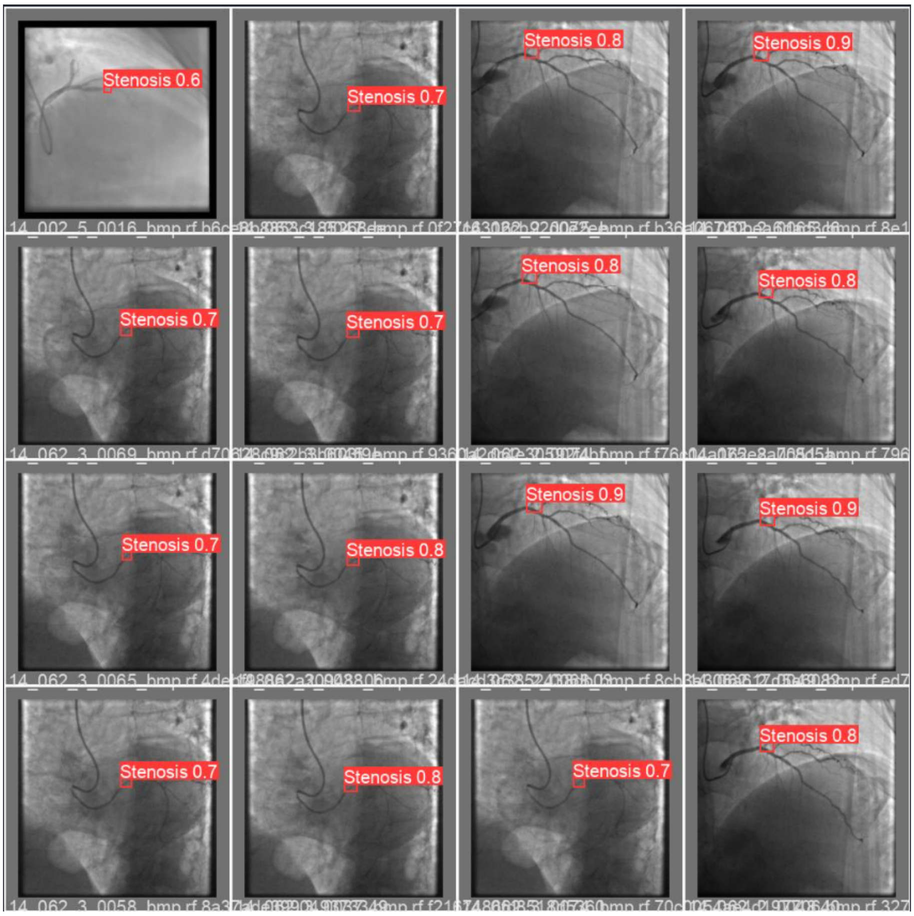

Figure 7.1: Pretrained YOLO Results

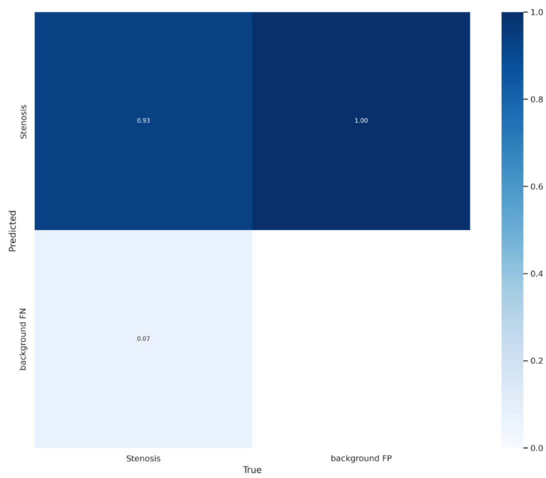

Figure 7.2: Confusion Matrix 1

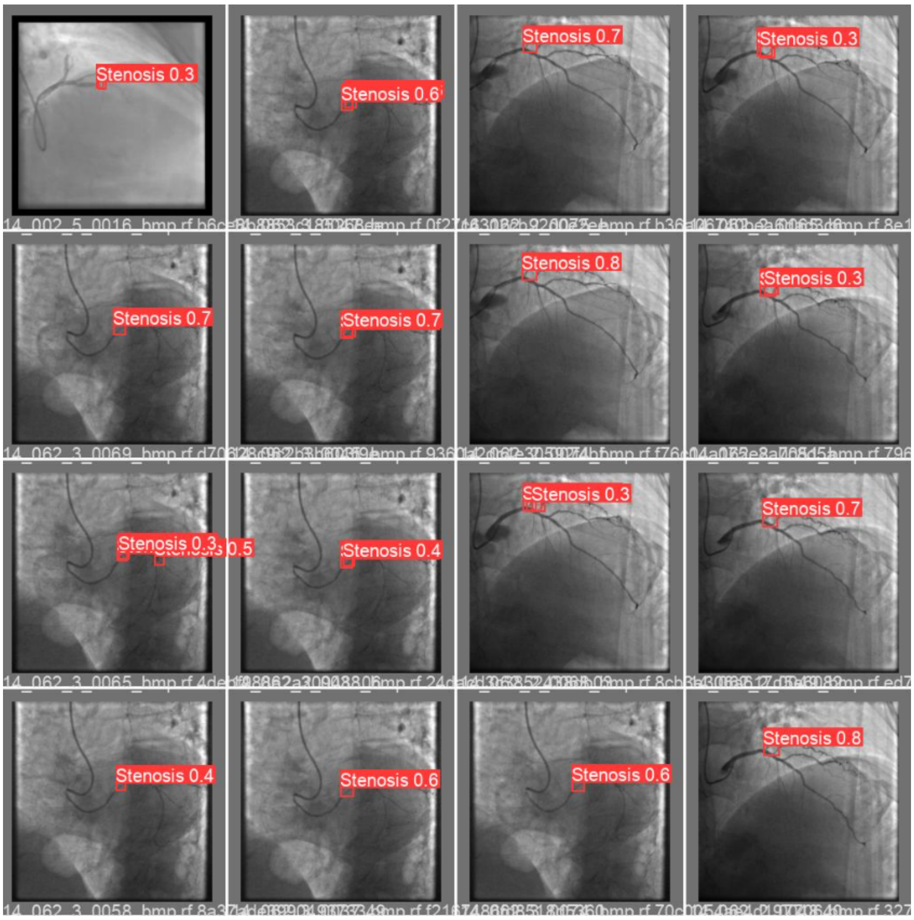

Figure 7.3: Transfer learning YOLO Results

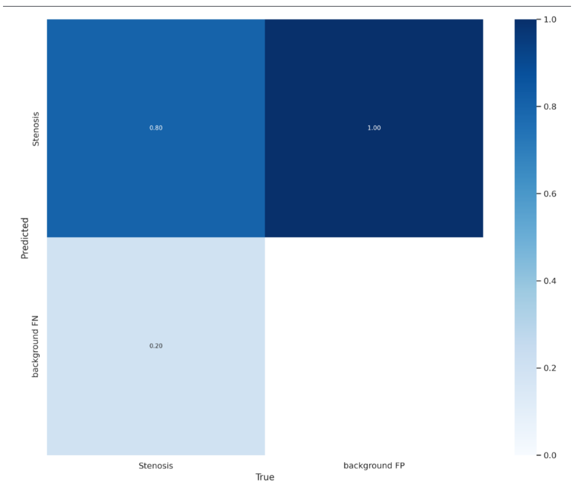

Figure 7.4: Confusion Matrix 2

Model	Metrics	
Sequential Model	IOU	0.918
	F1	0.91
VGG-16 (Transfer Learning)	IOU	0.9397
	F1	1.6667

The confusion matrix (7.4) shows the classification of true positives, true negatives, false positive and false negatives. The false positive value is 1.00 which means that stenosis was not predicted at the right place in a few images, known as type 1 error. However, the true positive value is 0.83, which means that the object was positive and predicted correctly. True negative is shown as blank since it is confirmed stenosis is present in each image. Finally, the false negative value is quite high at 0.20 which means that stenosis present in a few images was predicted to be negative, known as type 2 error.

Going through the confusion matrices for both YOLO models, the full YOLO model proves to be superior in performance with stenosis or medical-based dataset.

7.3 Sequential & VGG-16 Model

7.3.1 VGG-16 Model

The VGG-16 model performs extremely well for transfer learning and provides a good IOU prediction result, which can be seen below. This shows that the model works well with medical dataset even though it is trained with imageNet weights. The model was made to freeze the feature extraction layers and added its own Dense layers to the model with decreasing neurons towards the end. This model has proven to be a good indicator for performance in object detection cases. The images (7.5,7.6,7.7) are some of the best samples with the highest IOU prediction values. Within the metrics used, IOU, f1, and loss value is shown to be good as seen by the exemplary performance of the model predictions.

7.3.2 Sequential Model

The Sequential model performs extremely well for an untrained model and provides decent IOU rate, which can be seen below. This shows that the model works well with medical dataset even though not trained with weights. This model has proven to be a decent indicator for performance in object detection cases. The images (7.8,7.9) are some of the best samples with the highest IOU prediction values. Within the metrics used, IOU, f1, and loss value is shown to be good as seen by the exemplary performance of the model predictions.

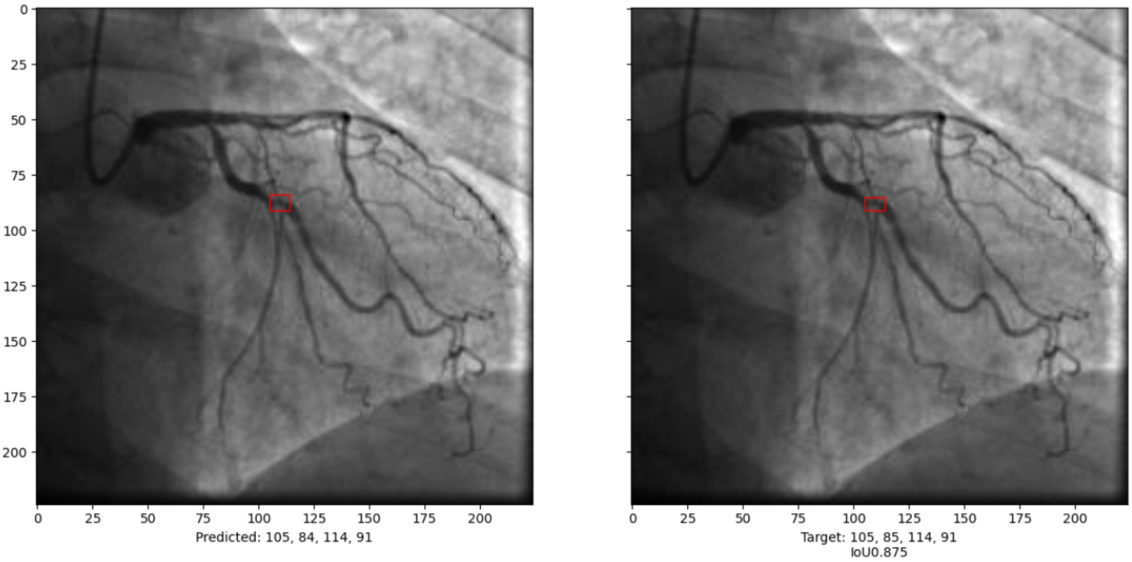

Figure 7.5: VGG Result 1

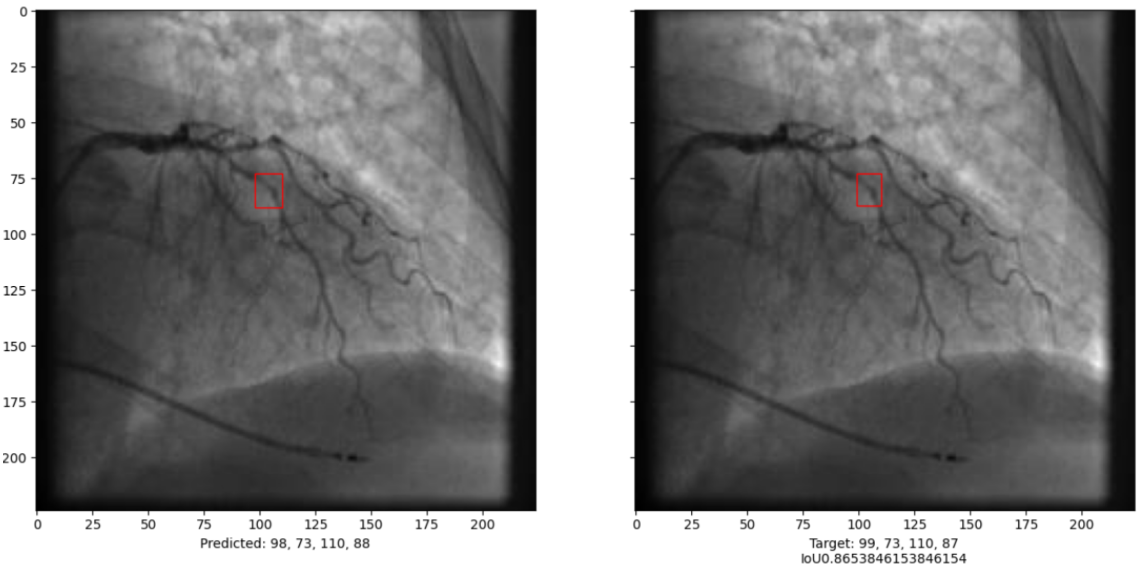

Figure 7.6: VGG Result 2

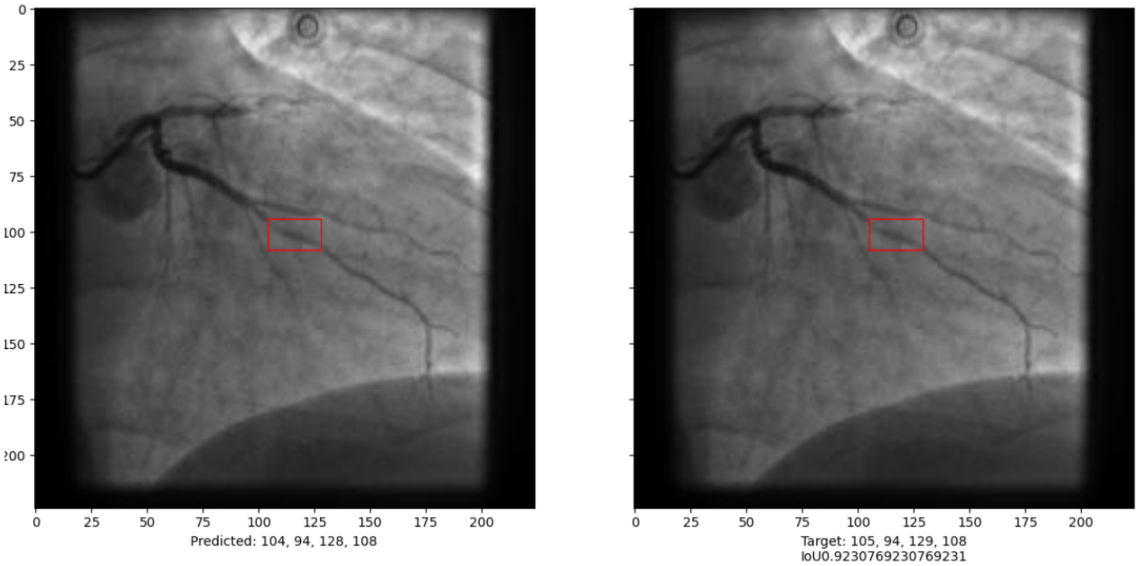

Figure 7.7: VGG Result 3

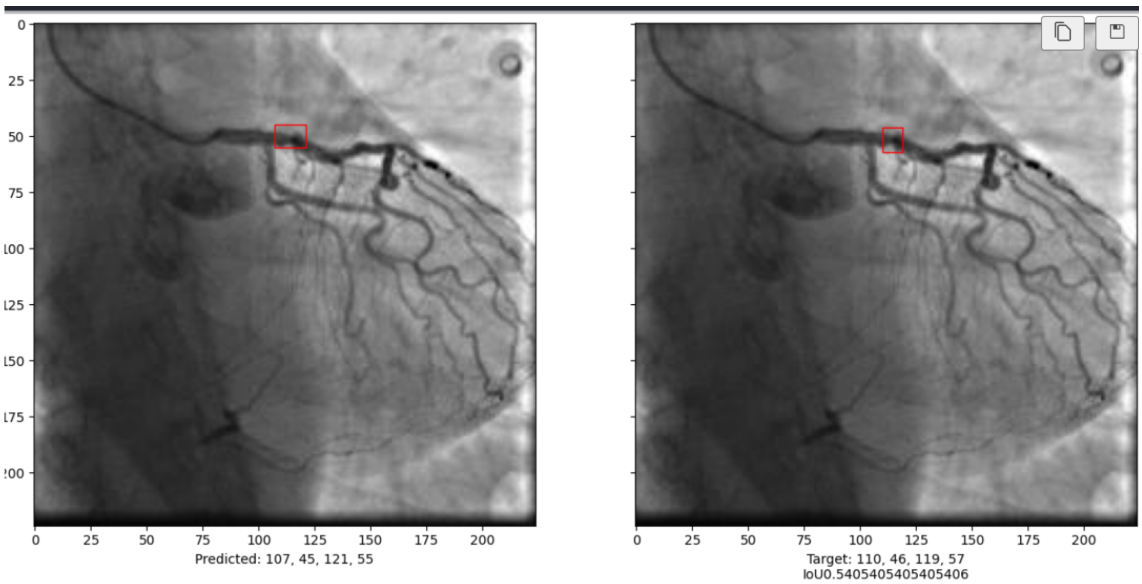

Figure 7.8: Seq Result 1

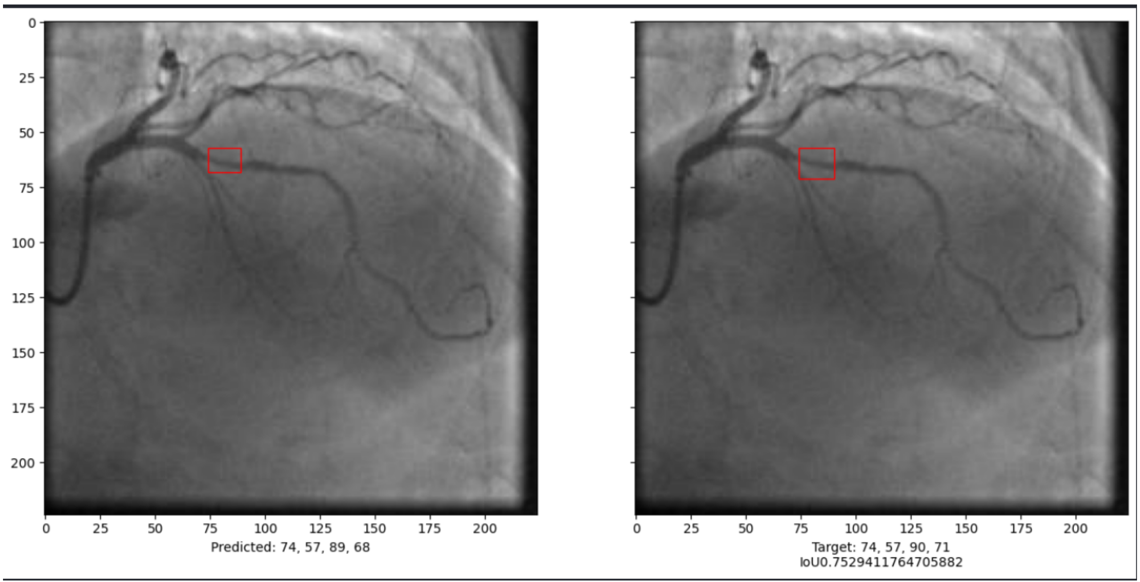

Figure 7.9: Seq Result 2

8 Conclusion & Future Work

The project has showcased the logic behind object detection and transfer learning techniques to aid pretrained and custom models for enhanced performance. We have shown VGG-16, YOLO model and our own custom based model with no pretrained weights. We have fulfilled the aims and objectives which are to understand how object detection can play a huge role in the field of healthcare. The use of deep learning models can prove a viable option for improved healthcare towards patients diagnosed with medical conditions.

We have also compared to the original paper approaches to see how our own preprocessing and model training will work with Python as compared to the original paper which used Amazon SageMaker. Highlighting the differences can help enlighten the reader to understand the problem at hand. The findings show that pretrained models prove to be superior in detecting stenosis accurately such as YOLO. Other pretrained models can easily outperform our own custom models as well, if trained long enough. This is important to understand how one must choose a model for trade-off in speed or accuracy.

Furthermore, improvements can be made to the experiments such as hyperparameter tuning, functional model, large dataset etc. Such are as follows.

8.1 Hyperparameter Tuning

Hyperparameter tuning refers to allowing the machine to find the right model architecture through exploration, given a set of commands. The questions solved can be how many layers to be added or what the right filter size will be for each layer. The hyperparameters are not the same as the model parameters, the model parameters are acquired through the training phase of the model. There are many methods of performing hyperparameter tuning such as grid search, bayesian optimization and random search. The grid search method is the basic method of hyparameter tuning which provides the option of creating machine learning models of every combination and thus, finding the best configuration to use. Random search is the opposite of grid search where the user provides a list of hidden values through which the machine will explore the best hyperparameters on its own through those values. Finally, bayesian optimization performs tuning differently, where models are tuned in parallel to share information with each other, therefore, making the tuning process faster and accurate (Jordan, 2017).

To improve model performance, hyperparameter tuning can be beneficial to find the best model fit for the stenosis dataset. However, the process of hyperparameter tuning requires additional setup and good knowledge of available parameters. This was not possible due to the short span of time for understanding the setup and parameters.

8.2 Functional API

Creating a Sequential model through using linear layers is not as efficient and accurate compared to using a functional model which is also called a resnet structure. A functional model will have a residual block of sharing layers and providing flexibility towards creating more complex algorithms or models (Faisal, 2020). Most pretrained models follow a functional structure. However, for the current stenosis dataset task, the functional model was not providing good results, therefore, a sequential model is used instead. However, improvement with functional models can play a big factor for object detection.

Bibliography

Stephen Allwright. How to interpret f1 score (simply explained), August 2022. URL https://stephenallwright.com/interpret-f1-score/.

James Beckerman. Heart disease: Types, causes, and symptoms, 2002. URL https://www.webmd.com/heart-disease/heart-disease-types-causes-symptoms.

BHF. What to expect in a cath lab: 360 degree video, 2020. URL https://www.bhf.org.uk/informationsupport/heart-matters-magazine/medical/watch-an-angiogram/what-to-expect-in-a-cath-lab.

Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection, 2020. URL https://arxiv.org/abs/2004.10934.

CDC. Trends in Infant Mortality Attributable to Birth Defects – United States, 1980-1995, 1998. URL https://www.cdc.gov/mmwr/preview/mmwrhtml/00054921.htm.

CDC. Coronary Artery Disease | cdc.gov, July 2021. URL https://www.cdc.gov/heartdisease/coronary_ad.htm.

X. Chen, Y. Fu, J. Lin, Y. Ji, Y. Fang, and J. Wu. Coronary artery disease detection by machine learning with coronary bifurcation features. *Applied Sciences*, 2020. doi: 10.3390/app10217656.

V. V. Danilov, K. Yu. Klyshnikov, O. M. Gerget, A. G. Kutikhin, V. I. Ganyukov, A. F. Frangi, and E. A. Ovcharenko. Real-time coronary artery stenosis detection based on modern neural networks. *Scientific Reports, [online]*, 11, 2021a. doi: 10.1038/s41598-021-87174-2.

Viacheslav Danilov, Alejandro Frangi, Kirill Klyshnikov, Anton Kutikhin, Evgeny Ovcharenko, and Olga Gerget. Angiographic dataset for stenosis detection. *Nature*, 1, 2021b. doi: 10.17632/ydrm75xywg.1. URL https://data.mendeley.com/datasets/ydrm75xywg/1.

DeepAI. Exploding gradient problem, May 2019. URL `https://deepai.org/machine-learning-glossary-and-terms/exploding-gradient-problem`.

D. J. DeNoon. *Congenital Heart Disease Explained*. WebMD, 2007. URL `https://www.webmd.com/heart-disease/guide/congenital-heart-disease`.

R. Donovan. *Everything You Need to Know about Heart Disease. [online] Healthline*. healthline, 2018. URL `https://www.healthline.com/health/heart-disease#symptoms-ofheart-disease`.

D. Duda-Pyszny, P. Trzeciak, and M. Gąsior. Coronary artery disease in women. *Polish Journal of Cardio-Thoracic Surgery, [online]*, 15(1):44–48, 2018. doi: 10.5114/kitp.2018.74675.

Rahmat Faisal. Keras model sequential api vs functional api, November 2020. URL `https://medium.com/analytics-vidhya/keras-model-sequential-api-vs-functional-api-fc1439a6fb10`.

Sabrina Felson. Atherosclerosis, 2021. URL `https://www.webmd.com/heart-disease/what-is-atherosclerosis`.

Aryan Garg. Use yolo v5 object detection algorithm for custom object detection, 2021. URL `https://bit.ly/3KNwR7f`.

GE Healthcare. Innova igs 530, 2020. URL `https://www.gehealthcare.co.uk/products/interventional-image-guided-systems/igs-for-cardiology/innova-igs-530`.

Jeremy Jordan. Hyperparameter tuning for machine learning models., November 2017. URL `https://www.jeremyjordan.me/hyperparameter-tuning/`.

Christopher J. Kelly, Alan Karthikesalingam, Mustafa Suleyman, Greg Corrado, and Dominic King. Key challenges for delivering clinical impact with artificial intelligence. *BMC Medicine*, 17(1):195, October 2019. ISSN 1741-7015. doi: 10.1186/s12916-019-1426-2. URL `https://doi.org/10.1186/s12916-019-1426-2`.

Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation, 2018. URL `https://arxiv.org/abs/1803.01534`.

Cara T. Mai, Jennifer L. Isenburg, Mark A. Canfield, Robert E. Meyer, Adolfo Correa, Clinton J. Alverson, Philip J. Lupo, Tiffany Riehle-Colarusso, Sook Ja Cho, Deepa Aggarwal, Russell S. Kirby, and National Birth Defects Prevention Network. National population-based estimates for major birth defects, 2010–2014. *Birth Defects Research*, 111(18):1420–1435, 2019. doi: https://doi.org/10.1002/bdr2.1589. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/bdr2.1589`.

P. Medrano-Gracia, J. Ormiston, M. Webster, S. Beier, C. Ellis, C. Wang, Ö. Smedby, A. Young, and B. Cowan. A study of coronary bifurcation shape in a normal population. *Journal of Cardiovascular Translational Research*, 10(1): 82–90, 2016. doi: 10.1007/s12265-016-9720-2.

S. Narula, K. Shameer, Salem Omar, J.T. A. M., Dudley, and P.P. Sengupta. Machine-learning algorithms to automate morphological and functional assessments in 2d echocardiography. *Journal of the American College of Cardiology*, 68 (21):2287–2295, 2016. doi: 10.1016/j.jacc.2016.08.062.

NHS. Angiography, 2017. URL `https://www.nhs.uk/conditions/angiography/`.

Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. Interpretml: A unified framework for machine learning interpretability, 2019. URL `https://arxiv.org/abs/1909.09223`.

World Health Organization. Cardiovascular diseases, 2019. URL `https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1`.

Ravindra Parmar. Common Loss functions in machine learning, September 2018. URL `https://towardsdatascience.com/common-loss-functions-in-machine-learning-46af0ffc4d23`.

W. C. A. Pereira, A. V. Alvarenga, A. F. C. Infantosi, L. Macrini, and C. E. Pedreira. A non-linear morphometric feature selection approach for breast tumor contour from ultrasonic images. *Computers in Biology and Medicine*, 40(11-12):912–918, 2010. doi: 10.1016/j.compbiomed.2010.10.003.

Y. Qu, X. Deng, S. Lin, F. Han, H. H. Chang, Y. Ou, Z. Nie, J. Mai, X. Wang, X. Gao, Y. Wu, J. Chen, J. Zhuang, I. Ryan, and X. Liu. Using innovative machine learning methods to screen and identify predictors of congenital heart diseases. *Frontiers in Cardiovascular Medicine*, 8, 2022. doi: 10.3389/fcvm.2021.797002.

Karthik Seetharam, Sirish Shrestha, and Partho P Sengupta. Cardiovascular imaging and intervention through the lens of artificial intelligence. *Interventional Cardiology 2021;16:e31.*, 2021. doi: 10.15420/icr.2020.04. URL `https://doi.org/10.15420/icr.2020.04`.

Martin G Seneviratne, Nigam H Shah, and Larry Chu. Bridging the implementation gap of machine learning in healthcare. *BMJ Innovations*, 6(2):45–47, 2020. ISSN 2055-8074. doi: 10.1136/bmjinnov-2019-000359. URL `https://innovations.bmj.com/content/6/2/45`.

Pranshu Sharma. Transfer learning | understanding transfer learning for deep learning, 2021. URL `https://www.analyticsvidhya.com/blog/2021/10/understanding-transfer-learning-for-deep-learning/`.

D. D. Strah, K. A. Kowalek, K. Weinberger, J. Mendelson, A. W. Hoyer, S. E. Klewer, and M. D. Seckeler. Worse hospital outcomes for children and adults with covid-19 and congenital heart disease. *Pediatric Cardiology*, 43(3):541–546, 2021. doi: 10.1007/s00246-021-02751-6.

Ren Jie Tan. Breaking down Mean Average Precision (mAP), March 2022. URL `https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52`.

Raman Venkat. Predicting Heart Disease Using Machine Learning? Don't!, 2020. URL `https://www.kdnuggets.com/predicting-heart-disease-using-machine-learning-dont.html/`. Section: 2020 Nov Opinions.

Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1571–1580. IEEE, 2020. ISBN 978-1-72819-360-1. doi: 10.1109/CVPRW50498.2020.00203. URL `https://ieeexplore.ieee.org/document/9150780/`.

X. Xu, T. Wang, Y. Shi, H. Yuan, Q. Jia, M. Huang, and J. Zhuang. Whole heart and great vessel segmentation in congenital heart disease using deep neural networks and graph matching. *Lecture Notes in Computer Science*, pages 477–485, 2019. doi: 10.1007/978-3-.

Su Yang, Jihoon Kweon, Jae-Hyung Roh, Jae-Hwan Lee, Heejun Kang, Lae-Jeong Park, Dong Jun Kim, Hyeonkyeong Yang, Jaehee Hur, Do-Yoon Kang, et al. Deep learning segmentation of major vessels in x-ray coronary angiography. *Scientific reports*, 9(1):1–11, 2019.

J. Zhang, S. Gajjala, P. Agrawal, G. H. Tison, L. A. Hallock, L. Beussink-Nelson, M. H. Lassen, E. Fan, M. A. Aras, C. Jordan, K. E. Fleischmann, M. Melisko, A. Qasim, S. J. Shah, R. Bajcsy, and R. C. Deo. Fully automated echocardiogram interpretation in clinical practice. *Circulation*, 138(16):1623–1635, 2018. doi: 10.1161/circulationaha.118.034338.