SEGMENTATION OF CORONARY ARTERIES AND ANALYSIS OF STENOSIS USING DEEP LEARNING TECHNIQUES

A PROJECT REPORT

Submitted By

A THAMIZH NAMBI 312215104014

POOJA PRIYA V 312215104069

POORVAJA S 312215104070

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SSN COLLEGE OF ENGINEERING

KALAVAKKAM 603110

ANNA UNIVERSITY :: CHENNAI - 600025

April 2019

ANNA UNIVERSITY : CHENNAI 600025

BONAFIDE CERTIFICATE

Certified that this project report titled **"SEGMENTATION OF CORONARY ARTERIES AND ANALYSIS OF STENOSIS USING DEEP LEARNING TECHNIQUES"** is the *bonafide* work of "**A THAMIZH NAMBI (312215104014)**, **POOJA PRIYA V (312215104069)**, and **POORVAJA S (312215104070)**" who carried out the project work under my supervision.

DR. CHITRA BABU	**DR. P. MIRUNALINI**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Professor,	Associate Professor,
Department of CSE,	Department of CSE,
SSN College of Engineering,	SSN College of Engineering,
Kalavakkam - 603110	Kalavakkam - 603110

Place:
Date:

Submitted for the examination held on............

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENTS

We thank GOD, the almighty for giving me strength and knowledge to do this project.

We would like to thank and deep sense of gratitude to our guide **DR. P. MIRUNALINI**, Associate Professor, Department of Computer Science and Engineering, for her valuable advice and suggestions as well as her continued guidance, patience and support that helped us to shape and refine our work.

My sincere thanks to **DR. CHITRA BABU**, Professor and Head of the Department of Computer Science and Engineering, for her words of advice and encouragement and I would like to thank our project Coordinator **DR. S. SHEERAZUDDIN**, Professor, Department of Computer Science and Engineering for his valuable suggestions throughout the project.

We express our deep respect to the founder **DR. SHIV NADAR**, Chairman, SSN Institutions. We also express our appreciation to our **DR. S. SALIVAHANAN**, Principal, for all the help he has rendered during this course of study.

We would like to extend our sincere thanks to all the teaching and non-teaching staffs of our department who have contributed directly and indirectly during the course of our project work. Finally, I would like to thank our parents and friends for their patience, cooperation and moral support throughout our life.

A THAMIZH NAMBI **POOJA PRIYA V** **POORVAJA S**

ABSTRACT

Cardiovascular disease (CVD) is a class of disease that involves the heart or blood vessels. Of the several diseases amongst CVD, Coronary Artery Disease (CAD) is the most common and lethal. We have proposed an automated diagnosis of CAD by analyzing the stenosis present in the coronary arteries. This is achieved using deep learning models for identifying the slices with coronary artery, segmenting the identified coronary arteries and by detecting and analyzing the stenosis. A combination of Convolutional Neural Network and Recurrent Neural Network (CNN-RNN) model is used to identify the 2D slices in which the coronary artery is present. The U-Net model then segments the coronary artery from identified slices. 3D reconstruction of segmented coronary artery is then done using Maximum Intensity Projection (MIP) and the 2D projection images are obtained from different viewing angles. From the 2D projection images, stenosis is detected and analyzed using U-Net. Experiments were conducted and performance was evaluated using the corpus created at different stages of the system.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Cardio Vascular Disease (CVD) is a general term for conditions affecting the heart or blood vessels. It is usually resulted by the formation of plaque inside the arteries (atherosclerosis) which increases risk of blood clots. Coronary Heart Disease (CHD), stroke, aortic disease and peripheral arterial disease are the common types of CVD. Amongst them, CHD is the most fatal resulting in over 7 million deaths by 2010. CHD occurs when plaque builds up in the arteries. When the plaque comes loose from the vessel wall, it clogs up the artery and reduces the amount of oxygen-rich blood getting to the heart which can be fatal. This condition of narrowing or clogging up of vessel is called stenosis and is the primary cause of CHD. The severity of stenosis depends on the percentage of blockage. Thus, early diagnosis of stenosis is very important to avoid fatal causes.

Although there are several imaging techniques to diagnose CHD, the gold standard technique is the Computed Tomography Angiography (CTA). Currently, stenosis detection is done by the medical experts, manually from the obtained 2D images of CTA. Since detecting CHDs from CTA is observer dependent, time consuming and error prone, a robust and efficient image segmentation procedure with quantitative shape analysis is the required to detect the presence of stenosis along the vessels. This system can be used as first level of filter for stenosis detection which assists the medical experts to analyze it further.

This project proposes an automatic segmentation of coronary arteries and detection of stenosis from the projection images obtained from the 2D axial slices of heart. Each of the 2D slices contain coronary arteries,non-coronary arteries such as pulmonary arteries and other parts such as aorta and veins. Since coronary artery is not present in every 2D slice of a patient, the slices containing coronary artery must be identified. This is achieved using a CNN-RNN model which also helps in predicting the starting and ending slices containing the coronary artery for a given patient. A U-Net model then segments the coronary arteries from the identified 2D slices. Since detection of stenosis from a single slice of the coronary artery is impossible, a 3D view of the coronary artery is constructed from the resulting images using which 2D projection images are obtained from different rotation angles in order to avoid the issue of vessel foreshortening. This helps in avoiding inaccurate measurements of vessels. The system then uses a U-Net model to analyse and detect the presence of stenosis in it.

CHAPTER 2

LITERATURE SURVEY

2.1 Segmentation of coronary artery

Segmentation using Tree Extraction, Frangi Vesselness Filter and Contour Detection was done in [2] . The Frangi filter measures the similarity to a tubular structure. But, the tree may not be automatically extracted caused by wrongly detected aorta.

A machine learning based Segmentation method for 3D CTA images, using Random Forests and Markov Random Field (MRF) optimisation was proposed in [10].

Segmentation propagation is the problem of transferring a Segmentation of an image to a neighbouring image in a sequence. The portions of the coronary artery have structures with a specific shape or appearance in each serial section slice of the heart, which only changes minimally from slice to slice. The coronary artery portions also exhibit specific inter-structure topology that constrains their neighbouring relations.

An automatic coronary extraction using Watersheds was proposed in [8] . It combined a recursive sequential tracking algorithm and morphological tools of homotopy modification and watersheds for extraction.

A propagation framework formulated as an optimal labelling problem solved using the graph-cut algorithm was implemented in [19]. The framework provides both homomorphic propagation and a local non-homomorphism strategy.

Image sequence segmentation containing : Global Motion Compensation, robust frame differencing and Curve Evolution was implemented in [23].

Both the global consistency and possible local inconsistency of the 2D structural topology is considered by propagating the 2D segmentations from one slice to another, in [18]. Each step of this propagation is an optimal labeling problem having two steps: a global labeling and a local labeling.

Fully Convolutional Neural Networks with deep supervision for efficient volume-to-volume segmentation was used in [7]. However, the algorithm is limited to vessel wall segmentation and does not separate between lumen and vessel wall.

Segmentation with Adaptive Local Statistics in Conditional Random Fields Framework which is a probabilistic graphical model was used in [9].

The coronary artery position is determined using the whole image sequence instead of a single frame in [16]. The vessel structure position is determined and the moving foreground areas are detected. The stationary areas in the image, such as the spine, ribs, and diaphragms are ignored which leads to a better processing time.

The coronary angiography sequence is rigidly registered so that the area around the point of interest appears stable in [6] .It examines a particular point on the Artery tree over the cardiac cycle and extracts a section of the Artery of interest, models it as a polyline, and tracks it.

Estimating coronary tissue motion based on the classical Lucas-Kanade (LK) algorithm for optical flow was proposed in [5] . The OF vector field quantifies the amount of misalignment between two consecutive frames in a sequence of images.

2.2 3D Reconstruction of coronary artery

3D reconstruction of medical images is vital in visualising the abnormalities in the structure and orientation of the organ of interest.

The contours of a surface are identified and the contours in consecutive images are connected through triangles in [11].

A volume rendering technique based on Monto Carlo integration is presented in [4] where the image is constructed by projection of a point cloud created by normalised reconstruction of volume as a probability density function.

An algorithm, which is capable of interactively generating Maximum Intensity Projection images using parallel projection and templates is proposed in [14]

An algorithm called Marching Cubes was proposed in [12], which identifies the surface from a logical cube created from 8 pixels, 4 from each adjacent side. It creates a polygonal representation of constant density surfaces from a 3D data array.

2.3 Detection and Analysis of Stenosis

A Recurrent Neural Network (RNN) for automatic detection and classification of coronary artery plaque and Stenosis was used in [21]. First, a 3D Convolutional Neural Network (CNN) is used to extract features along the coronary artery. Subsequently, the extracted features are aggregated by an RNN that performs multi-class classification tasks.

Learning based Stenosis detection using Support Vector Machine (SVM) was done in [17]. A multi-scale descriptor is extracted for every artery point and a decision algorithm learned by SVM based on this descriptor is used.
A Fuzzy Distance Transform (FDT) based approach to detect and quantify Stenosis was used in [20]. The stenoses were detected and analysed using the FDT values along the medical axis of an arterial tree obtained by skeletonization.

Deep Convolutional Neural Networks based stenosis detection was done in [1]. Several CNNs with different configurations by varying the number of convolution and pooling layers were trained and compared.

In [22], CNN is used to segment LV Myocardium from CCTA images. The segmented Myocardium is subsequently encoded using unsupervised Convolutional Auto Encoder (CAE) and SVM is used to classify patients based on features extracted from encodings.

Vessel enhancement diffusion filter along with Morphological operations are used in [13] to segment DICOM images of CTA scans. Vessel Centreline extraction and Vessel diameter estimation are carried out for finding the presence of Stenosis.

The application of two neural network algorithms for the detection of Coronary Artery Disease (CAD) is addressed in [3] from data obtained from thallium-201 (TL-201) Scintigraphy. It involves Back Propagation with Delta rule as the learning rule and Kohonen algorithm which uses an unsupervised learning rule to cluster data.

CHAPTER 3

SYSTEM REQUIREMENTS

This chapter discusses the hardware requirements, the software components used and methodologies for implementing the proposed system.

3.1 Hardware Requirements

- **Processor:** 4x Intel(R) Core(TM) Xeon CPU @ 1.70GHz
- **GPU:** NVDIA GTX 1080 Ti
- **GPU MEMORY:** 12GB
- **RAM:** 32 GB
- **Hard disk drive:** Seagate 1.0TB HDD (ST1000LM024 HN-M SCSI Disk Device)

3.2 Software Requirements

- OpenCV 3.4.0
- Scipy 1.0.1
- Numpy 1.14.2
- Matplotlib 2.2.2

- Sklearn 0.19.1
- Python 3.5.2
- Keras 2.2.4
- Tensorflow 1.7.0

3.3 Operating System

- Ubuntu 16.04 LTS

CHAPTER 4

METHODS AND ALGORITHMS USED

The project proceeds in three phases namely identification and segmentation of coronary artery, construction of 3D view of the coronary artery and analysis to detect the presence of stenosis in them. This chapter mentions the methods and algorithms used in those various stages of the project.

4.1 Contour Detection

A contour is a curve joining all the continuous points along the boundary of an object. The pixels inside a contour have similar colour or intensity. Identifying contours is a useful tool for shape analysis and object detection and recognition.

Segmentation of the coronary artery from the 2D axial slices is done by examining the contours present in the image. OpenCV functions are utilized to identify such contours and draw them. Each of the axial images are considered sequentially and segmentation of the coronary artery in them is done individually. Each slice is converted into binary image by applying a threshold or canny edge detection techniques.

The three OpenCV functions findContours, drawContours and pointPolygonTest are used in this project for contour detection and extraction.

4.1.1 Find Contours

This function is used to identify the different contours present in a binary image using border following technique. The arguments of this function include the source image, contour retrieval mode and contour approximation method. The source image is an 8-bit single channel binary image. Contour retrieval method corresponds to the hierarchical relationship to consider while identifying the contour points. The contour approximation method mentions if the points obtained must be compressed or encoded. The resultant contours of the source image are returned by representing each contour as an array of vector points.

4.1.2 Draw Contours

Once the contours are identified by the findContours function, the obtained contours are drawn on the source image using the drawContours function. Drawing the Contours on the source image enables easier selection of coronary artery blobs from the input image. This function takes the input image along with the Contour vector points as arguments. Additional arguments include thickness, colour and type of line needed to draw the contours on the image.

4.1.3 Point Polygon Test

This function is used in collaboration with mouse listeners. The role of the function is to determine whether a selected point is inside, outside or lies on the edges of a contour. The function returns a positive value if the selected point is inside a defined contour, a negative value if outside and zero value if its on the edges.

4.2 CNN-RNN

In the 2D slices of the patient, not every slice is found to contain the coronary artery. In order to predict the slices where the coronary artery is present, a combination of Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) is proposed. The hybrid network consists of a CNN and a Long Short Term Memory (LSTM) network connected in series. LSTM is a type of RNN which is best suited for learning long term dependencies in sequential data. The coronary artery is spread across various consecutive axial 2D slices which can be tracked sequentially. The LSTM is capable of processing sequential data but is not good at extracting features from image data. CNN is proved to be successful in extracting features of the image. Thus the CNN and LSTM are combined to process the sequential image data. The features are extracted by the CNN which are then aggregated by an LSTM which performs classification and says whether the slice contains the coronary artery or not, using which we can detect the starting and ending points of coronary artery.

4.2.1 CNN

A Convolutional Neural Network (CNN) is a type of deep, feed-forward artificial neural network that has been successful in analyzing visual imagery. It consists of an input layer, an output layer as well as hidden layers. There are 3 basic components in a basic CNN.

4.2.1.1 Convolutional Layer

Convolutional layers apply convolution operation to the input, passing the result to the next layer. The convolution emulates the response of an individual neuron in visual stimuli. We define a weight matrix which extracts certain features from the images.
This weight matrix shall now run across the image such that all the pixels are covered at least once, to give a convolved output. The values are obtained by adding the element wise multiplication values of the weight matrix and the highlighted part of the input image.
The weight matrix behaves like a filter in an image extracting particular information from the original image matrix. A weight combination might be extracting edges, while another one might be a specific colour, position, noise and so on.

4.2.1.2 Pooling Layer

Sometimes when the images are too large, we need to reduce the number of trainable parameters. It is then desired to periodically introduce pooling layers between subsequent convolutional layers. Pooling is done for the soul purpose of reducing the spatial size of the image. Pooling is done independently on each depth dimension; therefore, the depth of the image remains unchanged. The most common form of pooling applied is the max pooling.

4.2.1.3 Dense Layer

A dense layer is just a regular layer of neurons in a neural network. Each neuron receives input from all the neurons in the previous layer, thus densely connected. Dense layers perform classification on the features extracted by the convolutional layers and downsampled by the pooling layers. In a dense layer, every node in the layer is connected to every node in the preceding layer. Neurons in a dense layer have connections to all activations in the previous layer.

4.2.2 RNN

Recurrent Neural Network (RNN) is a type of neural network where the output from previous step is fed as input to the current step unlike traditional neural networks where all the inputs and outputs are independent of each other.

A usual RNN has short term memory but Long Short Term Memory (LSTM) allows RNN to retain their inputs over a long period of time. LSTM is a special variant of RNN which is best suited for learning long term dependencies. In an LSTM you have three gates - input, forget and output gate. These gates determine whether or not to let new input in (input gate), delete the information because it is not needed (forget gate) or to impact the output (output gate). 200 CTA slices of dimension 512x512 of one patient is considered as a sequential data. Here, the LSTM performs classification over the time series data i.e. whether it contains the coronary artery or not. LSTM is used since the temporal dynamics that connect the data is more important than the spatial content of each slice.

4.3 U-Net

The deep learning based U-Net model was originally proposed in [23]. U-Net architecture is separated into 2 symmetric parts: the contracting or downsampling path and the expansive or upsampling path.

4.3.1 Contracting or Downsampling path

The contracting path follows the typical architecture of a convolutional network. The original architecture of U-Net consists of the repeated application of two 3x3 convolutions, each followed by a Rectified Linear Unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. At each downsampling step, number of feature channels are doubled.

4.3.2 Expansive or Upsampling path

Every step in the expansive path consists of an upsampling of the feature map followed by a 2x2 convolution (up-convolution) that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU.
The cropping is necessary due to the loss of border pixels in every convolution.
At the final layer a 1x1 convolution is used to map each 64-component feature vector to the desired number of classes. In total the network has 23 convolutional layers.

A general architecture of U-Net is represented in Figure 4.1

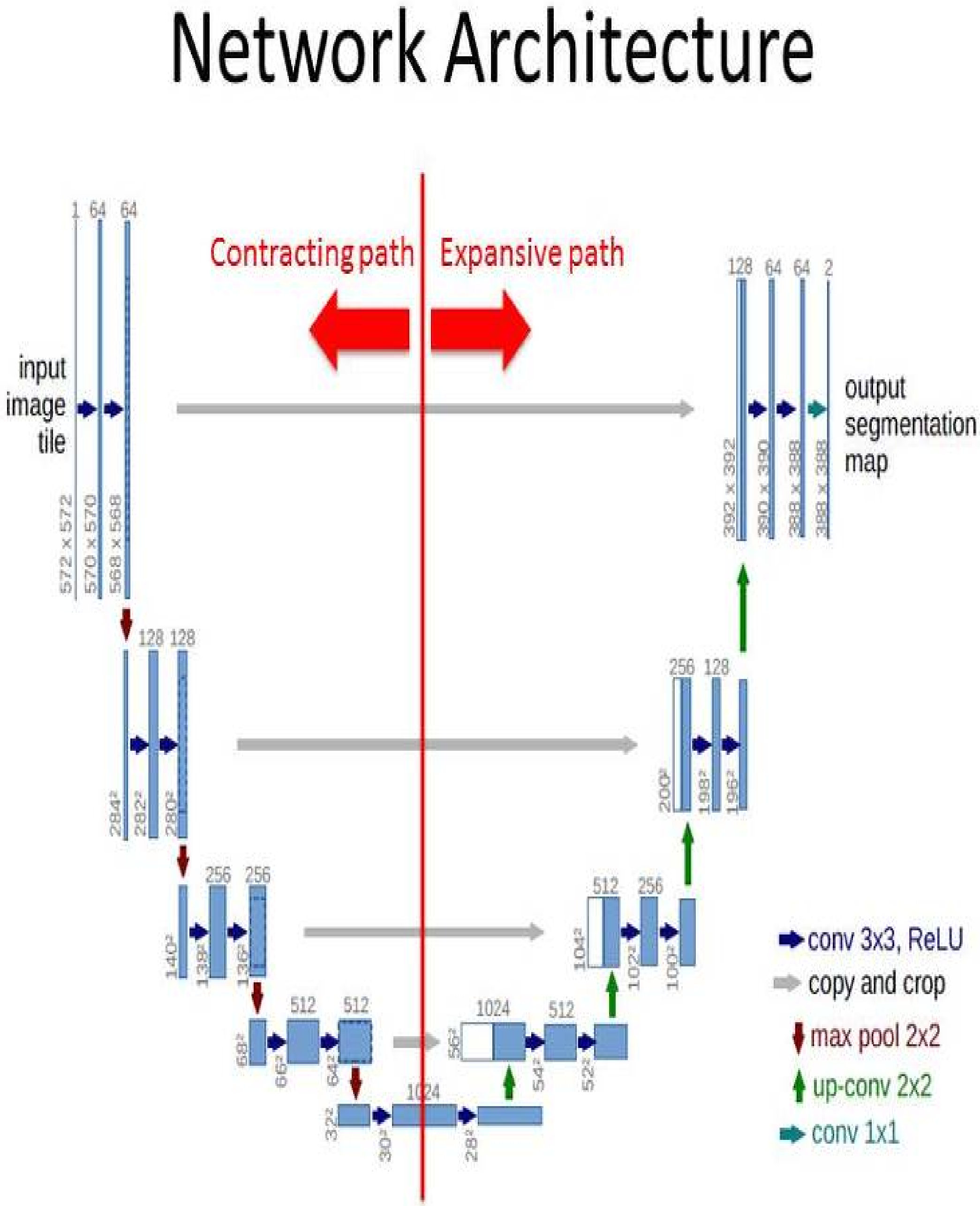

FIGURE 4.1: General architecture of U-Net

4.4 Maximum Intensity Projection

For the 3D reconstruction of the coronary artery from the 2D axial slices, the Maximum Intensity Projection (MIP) algorithm is used. This algorithm projects the voxel with maximum intensity in the parallel rays traced from the viewing point to the projection plane. The subsequent 2D axial slices from the segmentation model are stacked to form a z-stack. An n-dimensional numpy array is then constructed from the z-stack. MIP is then applied to get the 2D projection image of the coronary artery. The image is then rotated by 10 degrees along the XZ plane to obtain 36 such projection images to be fed for analysis.

CHAPTER 5

PROBLEM DEFINITION AND PROPOSED SYSTEM

5.1 Problem Statement

This project proposes a coronary artery segmentation and stenosis detection system from the 2D axial slices of heart obtained from CTA. Since coronary artery is not present in every 2D slice of a patient, the slices containing coronary artery must be identified. This is achieved using a CNN-RNN model which also helps in predicting the starting and ending slices of coronary artery for the given patient. Then, a U-Net model segments the coronary arteries from the identified 2D slices. Since detection of stenosis from a single slice of the coronary artery is impossible, a 3D view of the coronary artery is constructed from the resulting images using which 2D projection images are obtained from different rotation angles. The system then uses another U-Net model to detect the presence of stenosis and to analyse it.

5.1.1 Motivation

The detection of stenosis from examination of CTA by the medical experts is observer dependent, tedious and time consuming. The most of the algorithms currently used to automate stenosis detection are machine learning based. Feature engineering is the most important part for machine learning, which if not done properly, will affect the accuracy badly. Thus, we need to go for deep learning models which can learn the required features automatically from the input images. Moreover, the existing algorithms process the 2D projection images of the entire heart which is inefficient since unnecessary portions such as aorta and non-coronary arteries are also processed. This unnecessary processing can be avoided by removing all but the coronary artery portions from the axial 2D slices and then creating 2D projections of only the coronary artery. In this study, we have proposed an automated diagnosis of stenosis by analysing the 2D projection images of coronary artery obtained by segmenting the 2D axial slices using deep learning.

5.1.2 Overview

The idea of the project is to detect the presence of stenosis from the coronary arteries. Multiple 2D axial slices of the patient's heart are obtained from CTA. Each axial slice gives the view of interior tissues and arteries of the heart at different vertical layers. The entire set of axial images of a patient act as the input data for the subsequent model.

A CNN-RNN model is implemented to classify the input axial images based on the presence of coronary artery in them. The model performs a sequence based classification to select only those images which have a part of coronary artery. This method leverages the similarity between subsequent axial slices of the heart and enhances the accuracy of segmentation model. The output of this model is sent to the U-Net segmentation model for extracting the coronary artery in the slices.

A Deep learning U-Net architecture is used to segment the coronary artery from obtained 2D CTA slices. Each of the 2D slices contain coronary arteries, non-coronary arteries such as pulmonary arteries and other parts such as aorta and veins. Masking of the coronary artery in each slice is done to distinguish the coronary artery from other parts.

Since each slice depicts only a small portion of the coronary artery, a single 2D segmented image is insufficient to detect the stenosis. Multiple segmented images of the patient are gathered, and a 3D view of the coronary artery is constructed using 3D reconstruction algorithm. 2D projection images from this 3D structure are obtained for every 10 degree rotation resulting in 36 2D projection images per patient.

A U-Net based deep learning architecture is implemented on the projection images of the 3D view and its masks for detection and analysis of stenosis. Masks of the projection images are generated to specify the location and size of the stenosis, if present. The masks created help the U-Net to learn the features for detection of stenosis.

The objective of the present work is to extract and investigate every part of the coronary artery for stenosis and decrease the rate of false negatives over conventional manual methods.

The proposed method consists of the following steps as mentioned in Figure 5.1

1) Preparation of Ground Truth

2) Segmentation of Coronary Arteries

3) Reconstruction of 3D model

4) Analysis and Detection of Stenosis

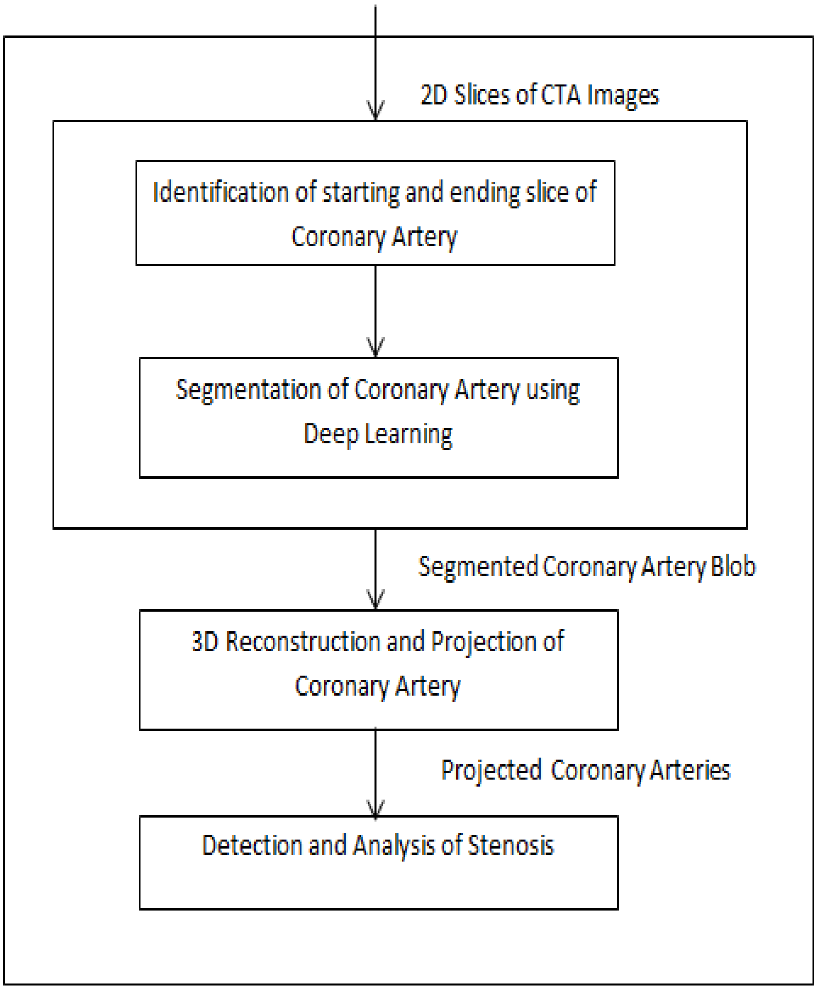

FIGURE 5.1: Overview of the system

5.2 Preparation of Ground Truth:

Deep learning algorithms are supervised and hence ground truth preparation is done to train the system. The coronary arteries which supplies blood to the heart, wrap around the outside of the heart with small branches diving into the heart muscle. The portions of the coronary artery in each subsequent slice have structures with a specific shape or appearance which only changes minimally from slice to slice.

The coronary artery portions also exhibit specific inter-structure topology that constrains their neighbouring relations. This property helps in accurate segmentation of the series of 2D serial-sectioned heart images using sequence segmentation. Sequence segmentation techniques for deep learning are studied and analysed for applicability and feasibility.

Creating the ground truth for the system involves obtaining multiple 2D heart slices and creating masks to distinguish the coronary artery from non-coronary artery in each slice.

Masks containing coronary artery are identified and extracted from the 2D heart slices using contour detection and extraction method with the help of expert guidance. The masks created contains only the coronary artery portion in each 2D heart slice in a black background.

5.2.1 Ground Truth Preparation Tool

To facilitate extraction of coronary artery blobs from input axial slices, we created a tool using Python and OpenCV. This tool is built by considering the input image as a collection of contours. Contour based extraction methods are used to identify contours, draw them and extract the chosen contour to create the mask.

The mask of an image refers to a plain black image except for the region which corresponds to the coronary artery portion. This is used by the models to understand the location, size, intensity and other parameters about the coronary artery present in an input slice.

The three OpenCV functions for contours used in this tool are findContours, drawContours and pointPolygonTest.

5.2.1.1 Find Contours

This function is used to identify the different contours present in a binary image using border following technique. The arguments of this function include the source image, contour retrieval mode and contour approximation method. The resultant contours of the source image are returned by representing each contour as an array of vector points.

```
Thresh = cv2.threshold(imgray,180,200,1)
contours = cv2.findContours(Thresh,cv2.RETR_LIST,cv2.CHAIN_APPROX_NONE)
```

5.2.1.2 Draw Contours

Once the contours are identified by the findContours function, the obtained contours are drawn on the source image using the drawContours function. Drawing the contours on the source image enables easier selection of coronary artery blobs from the input image.

```
contours = cv2.findContours(thresh,cv2.RETR_LIST,cv2.CHAIN_APPROX_NONE)
input_image= cv2.drawContours(img, contours, -1, (0,255,0), 1)
```

5.2.1.3 Point Polygon Test

This function is used in collaboration with mouse listeners. The role of the function is to determine whether a selected point is inside, outside or lies on the edges of a contour.

```
contours = cv2.findContours(Thresh,cv2.RETR_LIST,cv2.CHAIN_APPROX_NONE)
countour_id=cv2.pointPolygonTest(contours[i],(x_cord,y_cord),False)
```

Figure 5.2 refers to a 2D CT axial slice of a patient's heart. The extracted mask of the coronary arteries in the slice is given in Figure 5.3

FIGURE 5.2: Ground Truth

FIGURE 5.3: Mask

5.3 Segmentation of Coronary Arteries

Image segmentation is one of the most important processes of image processing. It is the technique of dividing or partitioning an image into parts, called segments. In this project, we propose to segment the coronary arteries from the 2D slices of the heart by removing the aortic regions using the deep learning architecture U-Net.

The U-Net architecture is built upon the Fully Convolutional Neural Network (FCNN) and modified in a way that it yields better segmentation in medical imaging.

The training data for the U-Net will be the CTA images along with the corresponding masks. The test data will comprise of another set of CTA images for which the network will predict the mask as output.

The data flow diagram for training the segmentation model is given in figure 5.4 depicting the data required and generated at each stage.

Figure 5.5 represents the data flow diagram for testing using the segmentation model.

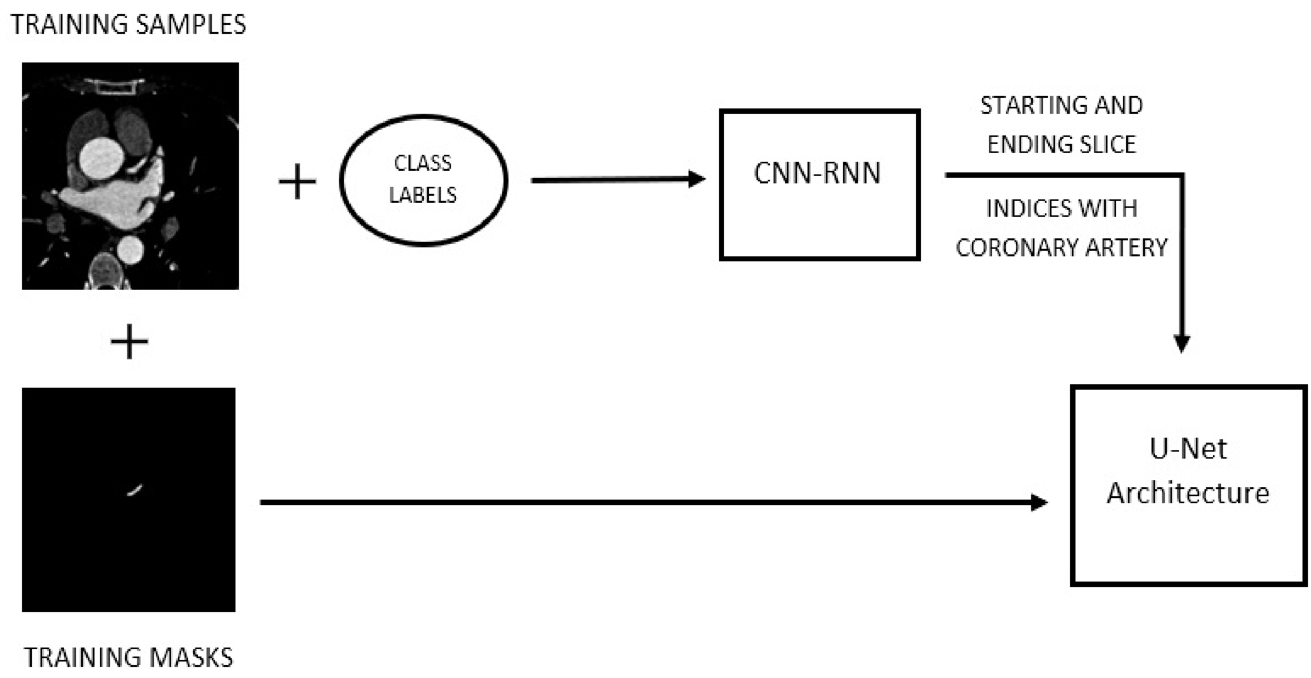

FIGURE 5.4: Data flow for training

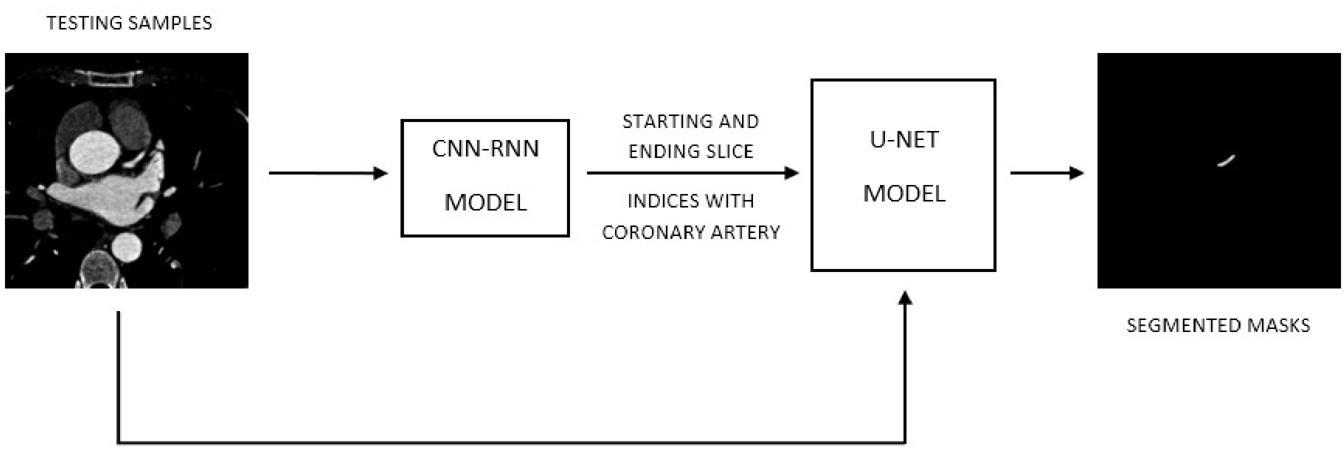

FIGURE 5.5: Data flow for testing

5.3.1 CNN-RNN model

5.3.1.1 Input

From the training dataset, depending on the ground truth prepared, labels for the are created for the training process. If the coronary artery masks of the input sample is completely black, it denotes the absence of coronary artery in the image. Hence, a 0 is as singed as the label for the image. Similarly, comparing the mask to a black image, it can be identified if the mask contains a portion of the coronary artery.In such a case, 1 is assigned to the image. The 2D slices, along with the labels are fed to the model.

5.3.1.2 Layers

The model accepts 200 CTA slices of dimensions 512x512, of each patient as a sequential input. The reason why a sequential model is used is because of the inter slice relationships among slices of the same patient.

The model consists of the following layers:

1. The contracting path has four time distributed 3x3 convolutional layers followed by 2x2 max pooling layers. The four convolutional layers have 32, 64, 96 and 128 filters respectively. The time distributed wrapper allows us to apply the layers to every temporal slice of the input.

2. A 3x3 ConvLSTM2D layer is then added, where the input transformations and recurrent transformations are both convolutional.

3. This is followed by a flattening layer which flattens the input without affecting the batch size.

4. The Expansive path has four dropout layers, each followed by a time distributed dense layer. The four dense layers have 256, 128, 64 and 1 filters respectively. The last layer is a 1x1 time distributed convolutional layer with sigmoid activation function.

Figure 5.6 depicts the CNN-RNN architecture used in the model.

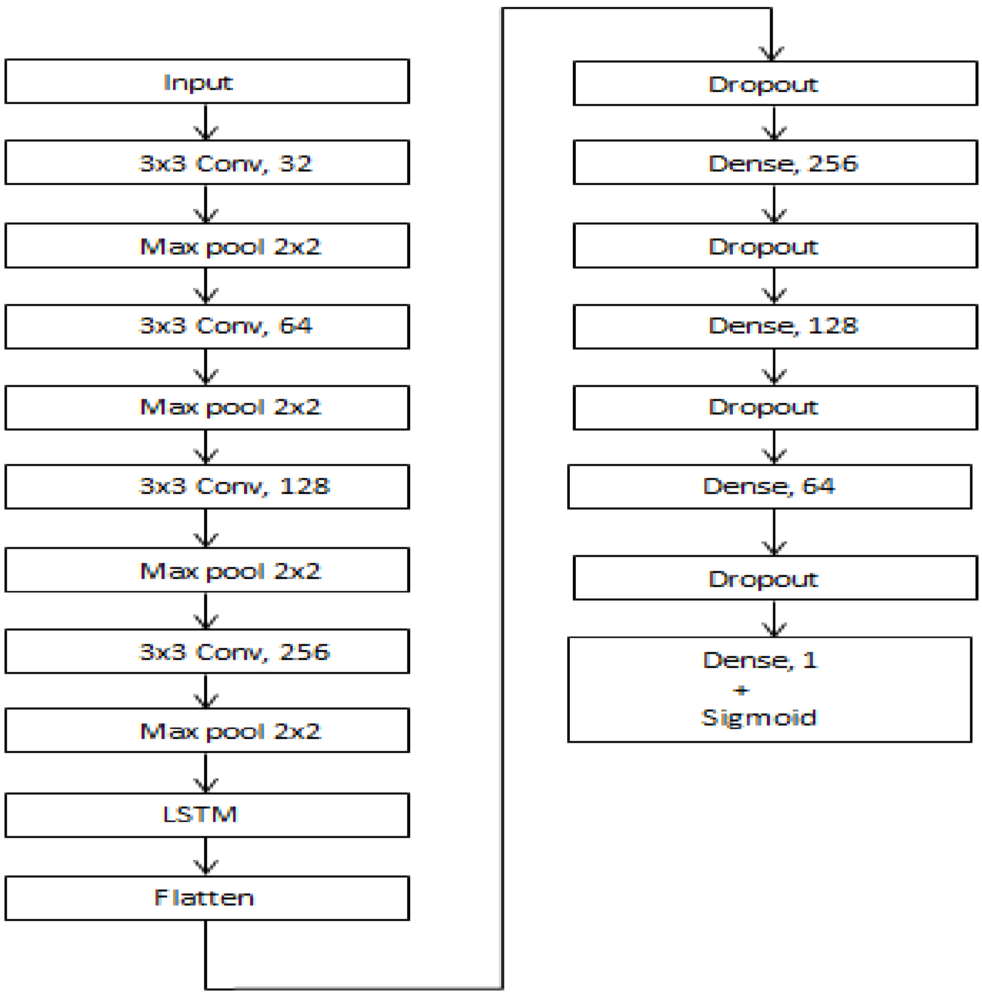

FIGURE 5.6: CNN-RNN architecture

5.3.2 U-Net Model

5.3.2.1 Input

The training dataset along with the masks prepared for the ground truth were fed to the U-Net model. The original size of the images is 512x512.

5.3.2.2 Layers

The model is built using blocks called residual blocks. Each residual block contains a 1x1 convolution layer, followed by a sequence of Batch normalization, ReLU activation function and a SeparableConv2D layer. This sequence is repeated twice in each block.

Batch Normalization normalises the output of each hidden layer which makes the layers independent of each other and helps in faster convergence. A ReLU activation function follows the Batch Normalization and makes the output non-negative. The SeparableConv2D layer performs a 3x3 Depth wise separable 2D convolution which can be viewed as a way to factorize a convolution kernel into two smaller kernels.

The model consists of the following layers:

1. The contracting path has six Residual blocks, each followed by a 2x2 max pooling. The six residual blocks have 32,64,96,128,256 and 512 filters respectively. The architecture of the residual block is given in Figure 5.7.

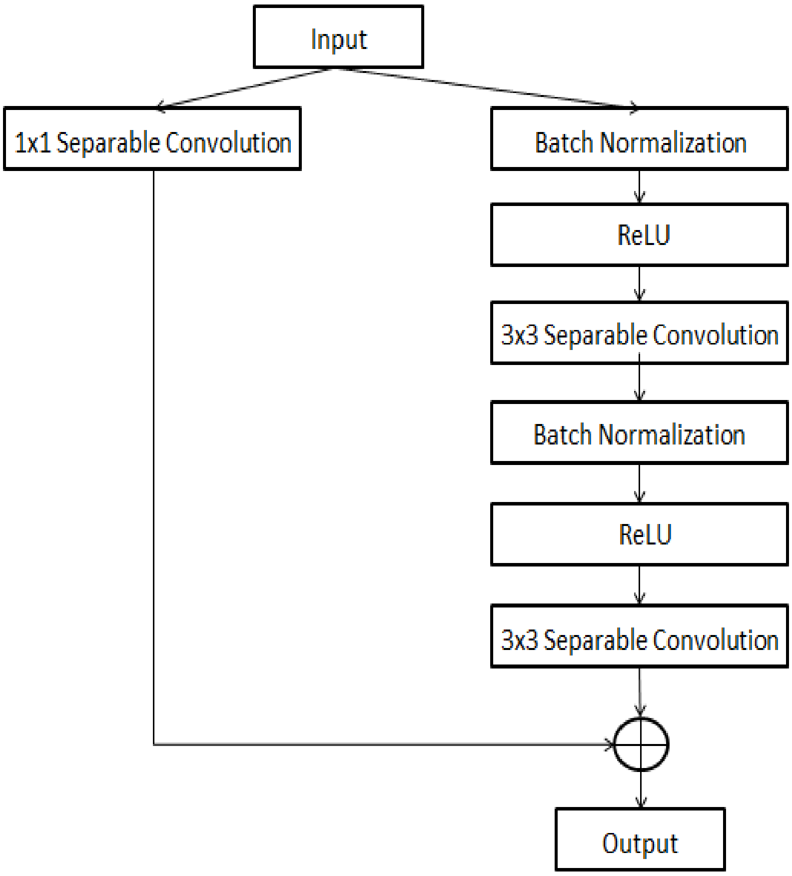

FIGURE 5.7: Residual block

2. A dropout layer is then added to generalize the model and to prevent overfitting. This is followed by a residual block of 512 filters.

3. The expansive path has six 2x2 up-sampling layers, each followed by a residual block. The six residual blocks have 512,256,128,96,64 and 32 filters respectively.

4. A dropout layer followed by a 1x1 convolution layer with sigmoid activation function.

The overall U-Net architecture of the proposed system is mentioned in Figure 5.8

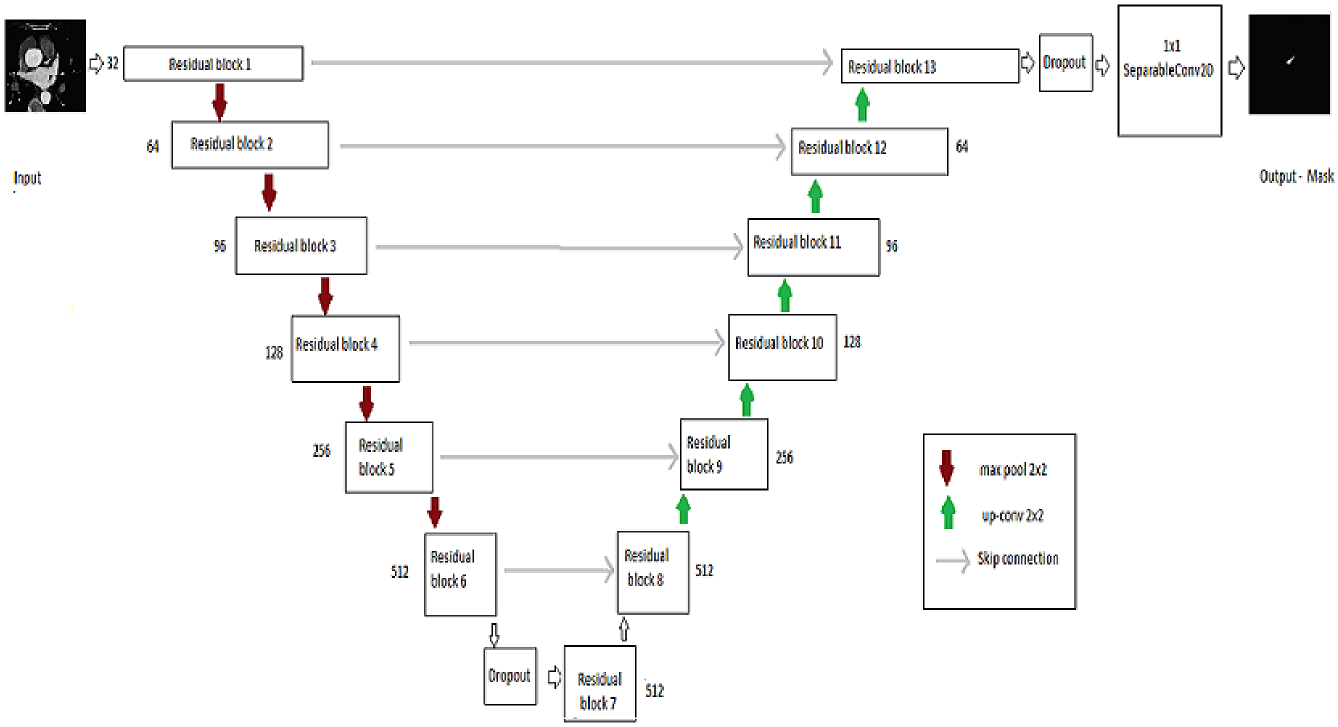

FIGURE 5.8: U-Net architecture

Figure 5.9 provides a sample input and output obtained from the U-Net model for segmentation of coronary arteries.

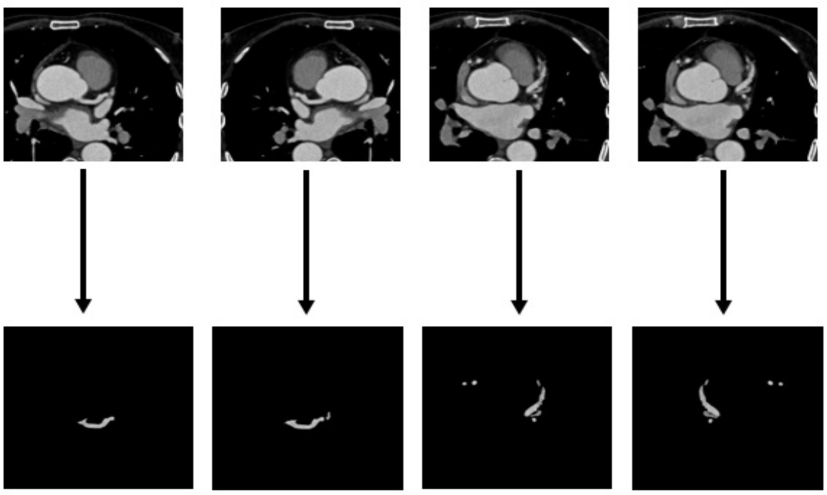

FIGURE 5.9: Input and output from U-Net

5.4 Reconstruction of 3D model

Due to the complexity involved in the visualisation of coronary artery in a single 2D axial image, a 3D view is constructed from the segmented images using Maximum Intensity Projection (MIP) algorithm. The absence of aortic and other irrelevant regions gives a better processing time for detection and analysis of stenosis.
The input for 3D reconstruction is the segmented axial slices containing the coronary artery. In 3D reconstruction, the segmented images are stacked to form a 3-dimensional view of the coronary artery. Refer Figure 5.10 for a projection image obtained from the 2D slices of the coronary artery.

5.4.1 Procedure

The resultant coronary artery masks from the U-Net model are binary images. These binary images are bitwise ANDed with the original image to get the segmented portion.

A z-stack is created from these images and is converted to a 3d numpy array.
The projection image is obtained through Maximum Intensiy Projection (MIP) algorithm applied on the 3D array. MIP is a reconstruction whereby in the view angle selected, the maximal intensity value along the line perpendicular to the view, represents this line of pixels in a 2D representation of the reconstructed body. The 2D projection images are created for every 10 degree rotation of the 3D image, resulting in 36 projection images for every patient.

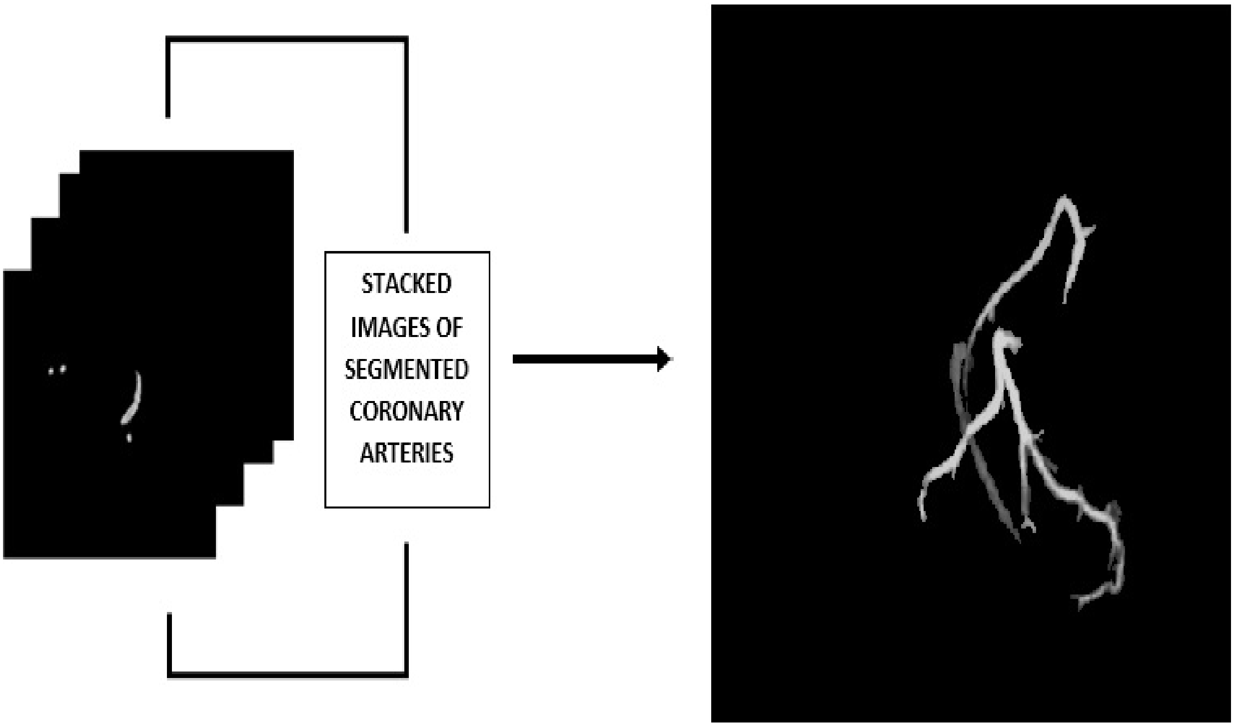

FIGURE 5.10: 3D reconstruction from segmented images using MIP

The Figure 5.10 shows the process of 3D reconstruction from segmented coronary artery images from input CTA images.

5.5 Detection and Analysis of Stenosis

The final stage of the project is to detect the presence of stenosis from the obtained 3D model of the coronary artery.

5.5.1 Ground Truth

The ground truth of the model is obtained by the 3D coronary artery generated for the patients from the slices by the previous models. For each patient, 3D projection images of the coronary artery are obtained for every 10 degree angle. So, 36 projection images of the coronary artery are obtained for each patient.

Using expert guidance and medical records, we had manually identified the region of stenosis in each slice for each patient. Correspondingly, a mask is created for every image where the location of the stenosis is depicted in a black background.

The below mentioned Figure 5.11 shows a projection image of the coronary arteries divided into grids to create the mask of stenosis.

The figure 5.12 mask of the stenosis extracted from the 2D projection image of the coronary artery.

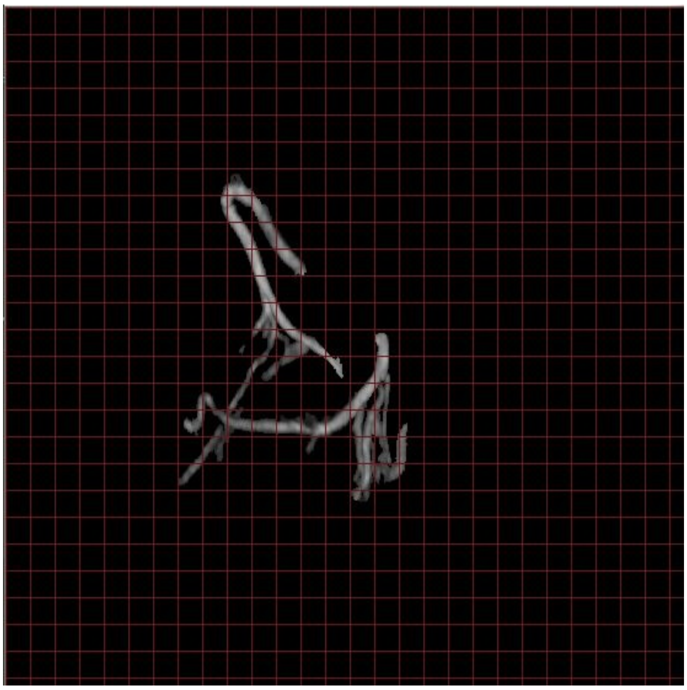

FIGURE 5.11: Projection image of coronary artery to create mask of stenosis

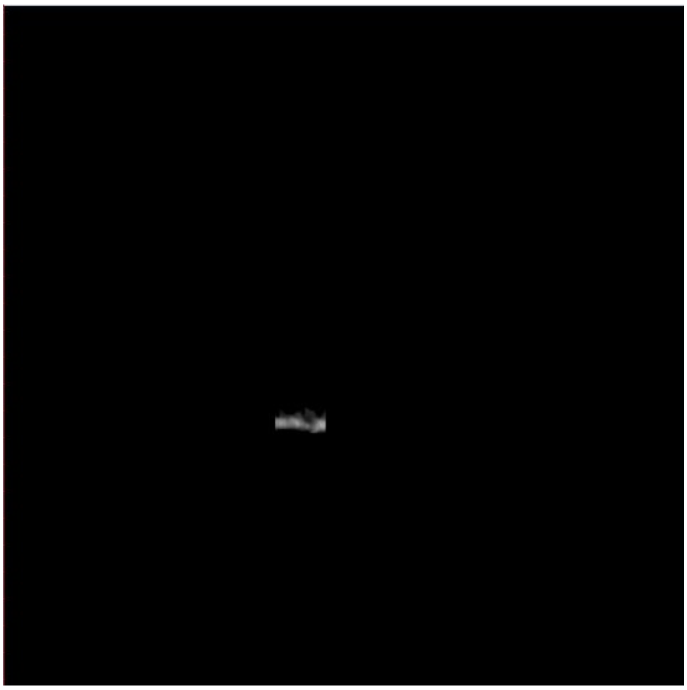

FIGURE 5.12: Mask of stenosis present in the coronary artery

CHAPTER 6

EXPERIMENTS AND RESULTS

The details of the dataset used in the project work, performance metrics used for the evaluation of experiments and results obtained are presented in this chapter.

6.1 Dataset Creation

The project involves creation of two different data sets namely 2D axial slices of CTA images and projection images of coronary artery obtained from 3D construction.

- 2D axial slices of heart has been used for classification and segmentation of coronary artery.
- Projection images of coronary artery has been used for detection and analysis of stenosis.

The 2-dimensional axial Computed Tomography Angiography (CTA) images of the heart are collected for 50 patients. Each patient has about 200-250 axial slices depicting the internal structure of the heart and arteries. The slices are collected from Bill Roth Hospital, Chennai. The images are obtained in DICOM format from the CTA scanner. They are converted to Tag Image File Format (TIFF) format by MicroDicom tool. Each image is normalised to be of 512x512 dimensions.

6.2 Performance Metrics

The performance metrics used in the system are based on the four outcomes of binary classification.

1. **True Positives TP** - Data points labeled as positive that are actually positive
2. **False Positives FP** - Data points labeled as positive that are actually negative
3. **True Negatives TN** - Data points labeled as negative that are actually negative
4. **False Negatives FN** - Data points labeled as negative that are actually positive

The performance of the system is evaluation using metrics like Precision (P), Recall (R), Accuracy (Acc) and Intersection over union (IoU).

- Accuracy can be calculated as the ratio of true results and total number of samples in the data.

$$\textbf{Accuracy (Acc)} = \frac{TruePositives + TrueNegatives}{Total}$$

- Precision is the ability of a classification model to return only relevant instances. It can be calculated as actual positives out of predicted positives.

$$\textbf{Precision} = \frac{TruePositives}{TruePositives + FalseNegatives}$$

- Recall is the ability of a classification model to identify all relevant instances. It is calculated as actual positives out of total positives.

$$\textbf{Recall} = \frac{TruePositives}{TruePositives + FalseNegatives}$$

- The performance of U-Net for segmentation of coronary artery is evaluated based on the parameter IoU (Intersection over Union), also referred to as Jaccard index. This essentially quantifies the percent overlap between the target mask and our prediction output. This metric is closely related to the dice coefficient. In other words, the IoU metric measures the number of pixels common between the target and prediction masks divided by the total number of pixels present across both masks.

$$\textbf{IoU} = \frac{Target \cap Prediction}{Target \cup Prediction}$$

6.3 Experiments Performed

The system consists of two major phases namely segmentation of coronary artery from 2D axial slices and analysis of stenosis from projection images of coronary artery. To ascertain the performance of our proposed system, the following experiments are conducted.

- **Experiment 1** - Identification of coronary artery slices.

- **Experiment 2** - Segmentation of coronary artery from 2D axial slices.
- **Experiment 3** - Detection and analysis of stenosis from projection images of coronary artery.

6.3.1 Identification of coronary artery slices using CNN-RNN

This step involves identification of those slices which has a portion of coronary artery in them. This is preferred to facilitate the segmentation process as only relevant slices are sent for further processing. This step accepts the input as the set of 2D axial slices obtained from CTA. The model used, classifies the input slices by identifying if it contains a portion of coronary artery in them. This information is then used to identify the starting and ending slice which has the coronary artery. Only those 2D axial slices within this range is sent for further processing, thus eliminating the need to process the entire set of input images.

A CNN-RNN model is implemented to classify the input 2D axial slices. In the input set of images, each subsequent image is similar to the previous one and has only minor changes in terms of structure and positioning. To leverage this property of the slices, the RNN is Incorporated in the model to analyse the input set of images as a sequence.

6.3.1.1 Results

The model is trained for 70 epochs with Binary Cross Entropy loss function and Adam optimizer for the training data set mentioned previously with 4 as batch size. CTA images of 40 patients are used as the training data for the model. The accuracy obtained with Binary Cross Entropy as loss function from the validation data was 0.9835. The overall loss value was found to be 0.0389.

To evaluate the performance of the model, we have used precision, accuracy and recall as the performance metrics. The results obtained are mentioned in the table 6.1.

TABLE 6.1: Results of CNN-RNN model

Patient Number	**TP**	**FP**	**TN**	**FN**	**ACCURACY**	**PRECISION**	**RECALL**
1	119	1	80	0	0.995	0.992	1.000
2	116	1	83	0	0.995	0.991	1.000
3	117	1	77	5	0.970	0.992	0.959
4	136	2	58	4	0.970	0.986	0.971
5	143	0	51	6	0.970	1.000	0.960
6	131	1	61	7	0.960	0.992	0.949
7	126	2	61	11	0.935	0.984	0.920
8	113	3	70	14	0.915	0.974	0.890
9	121	1	60	18	0.905	0.992	0.871
10	114	14	64	8	0.890	0.891	0.934

The CNN-RNN model has achieved a recall of 94.5%, precision of 97.9% and an accuracy of 95.1% using the test data. This shows that all coronary arteries are detected and only few non-coronary arteries are misclassified as coronary arteries. This misclassification is due to variation in position and size of the heart and difference in imaging quality, which makes it difficult in distinguishing the coronary from non-coronary arteries.

6.3.2 Segmentation of coronary artery from 2D axial slices using U-Net architecture

The previous model returns the starting and ending slice to consider for segmentation process. Based on that values, only those slices which are within the range are selected and sent to the segmentation model.

The segmentation model is based on U-Net architecture. Each image given as an input to the segmentation model is provided with a mask of the coronary artery in it. The mask corresponds to a black image which contains only the coronary artery region in the image at the exact position, size and intensity. The U-Net model understands to segment the artery by learning the images and the respective masks.

The output of this model is the set of masks corresponding to the input images. These masks represent only the coronary artery and are sent to the next model for the analysis of stenosis.

6.3.2.1 Results

The model is trained for 30 epochs for 40 patients with Mean Squared Error (MSE) loss function and Adam optimizer for the training data set mentioned previously with 4 as batch size. The accuracy obtained from the validation data was 0.9992 and the loss obtained was 0.00065.

TABLE 6.2: Results of U-Net segmentation model

Patient Number	IoU
1	0.6402
2	0.6036
3	0.6086
4	0.6537
5	0.5878
6	0.5975
7	0.6240
8	0.6271
9	0.6973
10	0.7666

The model was tested with 2D CTA slices of 10 patients to produce the corresponding masks . The performance of the model is evaluated by Insertion over Union (IoU) metric and the results are mentioned in table 5.2. It can be observed that the overall IoU score is 0.64. This indicates that there is a 64% overlap between the ground truth and the predicted masks on an average. It is observed that any IoU score which is greater than 0.5 is good enough for segmentation. Though this is a pretty good score, a few of the non-coronary artery parts were also segmented by the model, due to their size and shape being very similar to coronary arteries.

6.3.3 Detection and analysis of stenosis from projection images of coronary artery using U-Net architecture

This model aims in locating the stenosis in projection images obtained from 3D construction of the heart. The projections generated for the patients from the slices by the previous models along with the masks act as the ground truth of the model. For each patient, projection images of the coronary artery are obtained for every 10 degree angle. So, 36 projection images of the coronary artery are obtained for each patient.

The output of this model is the set of masks for each input projection image of the coronary artery. If stenosis is present and visible at the particular projection angle, the model generates the mask of the stenosis in a black background.

6.3.4 Results

1440 projection images of the artery along with their masks given as the input to the model are used to train the model. The model is run for 80 epochs with Binary cross entropy as the loss function.

```
Epoch 00072: loss did not improve from 0.00010
Epoch 73/80
745/745 [==============================] - 469s 630ms/step - loss: 9.5959e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9998

Epoch 00073: loss improved from 0.00010 to 0.00010, saving model to unet_analysis_80_bin.hdf5
Epoch 74/80
745/745 [==============================] - 470s 631ms/step - loss: 8.4974e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9998

Epoch 00074: loss improved from 0.00010 to 0.00008, saving model to unet_analysis_80_bin.hdf5
Epoch 75/80
745/745 [==============================] - 469s 629ms/step - loss: 7.6822e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9998

Epoch 00075: loss improved from 0.00008 to 0.00008, saving model to unet_analysis_80_bin.hdf5
Epoch 76/80
745/745 [==============================] - 470s 630ms/step - loss: 8.8914e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9998

Epoch 00076: loss did not improve from 0.00008
Epoch 77/80
745/745 [==============================] - 468s 628ms/step - loss: 8.7884e-05 - acc: 1.0000 - val_loss: 0.0036 - val_acc: 0.9998

Epoch 00077: loss did not improve from 0.00008
Epoch 78/80
745/745 [==============================] - 469s 630ms/step - loss: 8.9462e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9998

Epoch 00078: loss did not improve from 0.00008
Epoch 79/80
745/745 [==============================] - 469s 630ms/step - loss: 8.1803e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9998

Epoch 00079: loss did not improve from 0.00008
Epoch 80/80
745/745 [==============================] - 471s 632ms/step - loss: 8.2747e-05 - acc: 1.0000 - val_loss: 0.0037 - val_acc: 0.9997
```

FIGURE 6.1: Detection and analysis of stenosis

From Figure 6.1 it is observed that the loss evaluated by the model was 0.00008 with an accuracy of 99.99.

CHAPTER 7

CONCLUSION AND FUTURE WORK

In this thesis, we have discussed in detail the deep learning models for automating the segmentation of coronary artery from 2D axial slices of CTA images and for locating stenosis in the segmented coronary arteries.

In the proposed system, CNN-RNN, which is a combination of CNN and RNN is used to identify the slices where the coronary artery starts and ends. This is done by classifying every slice based on whether it contains coronary artery or not. The next deep learning model U-Net uses this information and segments the coronary artery from the slices identified. Then the 3D reconstruction algorithm Maximum Intensity Projection (MIP) is used to get the projection images from 36 different angles for a patient. Finaly U-Net is again used to locate the stenosis in projection images.

As future work, we can extend the system to include more information about stenosis such as severity and type of stenosis. Since stenosis can be calcified or non-calcified stenosis, a classifier can be built to identify the type of stenosis.

REFERENCES

1. Antczak, K., and Liberadzki, L. (2018) ‘Stenosis Detection with Deep Convolutional Neural Networks’, MATEC Web of Conferences. Vol. 210, pp. 4001.

2. Broersen, A., Kitslaar, P., Frenay, M., and Dijkstra, J. (2012) ‘Fast, Robust Extraction for the Nice Challenge on Coronary Artery Segmentation of the Tree’, In Proc. of MICCAI Workshop 3D Cardiovascular Imaging: a MICCAI Segmentation Challenge.

3. Cios, K. J., Goodenday, L. S., Merhi, M., and Langenderfer, R. A. (1989) ‘Neural Networks in Detection of Coronary Artery Disease’, IEEE Computers in Cardiology, pp. 33-37.

4. Csebfalvi, B. and Szirmay-Kalos, L. (2003) ‘Monte Carlo volume rendering’, IEEE Visualization, VIS 2003., Seattle, WA, USA, pp. 449-456.

5. Danilouchkine, M. G., Mastik, F., and van der Steen, A.F. (2009) ‘A study of coronary artery rotational motion with dense scale-space optical flow in intravascular ultrasound’, Physics in Medicine and Biology, No.6, pp. 1397.

6. Dubuisson-Jolly, M. P., Liang, C. C., and Gupta, A. (1998) ‘Optimal polyline tracking for artery motion compensation in coronary angiography’, Sixth International Conference on Computer Vision, pp. 414-419.

7. Fredrik, R. (2018) ‘Deep learning for Coronary Artery Segmentation in CTA images’, Chalmers University of Technology, Gothenburg, Sweden.

8. Haris, K., Efstratiadis, S. N., Maglaveras, N., Gourassas, J., Pappas, C., and Louridas, G. (1997) ‘Automated Coronary Artery Extraction using Watersheds’, IEEE Computers in Cardiology 1997, pp. 741-744.

9. Hu, Y. C. J., Grossberg, M. D., and Mageras, G. S. (2008) ‘Semi-Automatic Medical Image Segmentation with Adaptive Local Statistics in Conditional Random Fields Framework’, 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 3099-3102.

10. JDeng, J., Xie, X., Alcock, R., and Roobottom, C. (2014) ‘3D Interactive Coronary Artery Segmentation using Random Forests and Markov Random Field Optimization’, IEEE International Conference on Image Processing (ICIP), pp. 942-946.

11. Keppel, E. (1975) ‘Approximating Complex Surfaces by Triangulation of Contour Lines’, IBM Journal of Research and Development, Vol. 19, No. 1, pp. 2-11.

12. Lorensen, W. E., and Cline, H. E. (1987). ‘Marching Cubes: A High Resolution 3D Surface Construction Algorithm’, ACM SIGGRAPH Computer Graphics, Vol. 21, No. 4, pp. 163-169.

13. Mohan, N., and Vishnukumar, S. (2016) ‘Detection and Localization of Coronary Artery Stenotic Segments using Image Processing’, IEEE International Conference on Emerging Technological Trends (ICETT), pp. 1-5.

14. Mroz, L., Konig, A. and Groller, E. (1999) ‘Real-Time Maximum Intensity Projection’, Data Visualization’99, Proceedings of the Joint EUROGRAPHICS - IEEE TCCG Symposium on Visualization, Springer, Vienna, pp. 135144.

15. Ronneberger, O., Fischer, P., and Brox, T. (2015) ‘U-net: Convolutional Networks for Biomedical Image Segmentation’, International Conference on Medical Image Computing and Computer-assisted intervention, Springer, pp. 234-241.

16. Tenekeci, M. E., Pehlivan, H., and Kaya, Y. (2018) 'Improving performance of Coronary Artery Segmentation using Calculated Vessel Location from the Angiogram', Biomedical Research, Vol. 29, Issue 1, pp. 130-136.

17. Toledano, M., Lindenbaum13, M., Lessick46, J., Dragu, R., Ghersin56, E., Engel56, A., and Beyar246, R. (2010) 'Learning to Detect Coronary Artery Stenosis from Multi-Detector CT imaging', No. CS Technion report CIS-2010-02. Computer Science Department, Technion.

18. Waggoner, J. W., Simmons, J., and Wang, S. (2012) 'Image Sequence Segmentation Combining Global Labeling and Local Relabeling and its Application to Materials Science Images', Computational Imaging X International Society for Optics and Photonics, Vol. 8296, pp. 829606.

19. Waggoner, J., Zhou, Y., Simmons, J., De Graef, M., and Wang, S. (2013) '3D Materials Image Segmentation by 2D Propagation: A Graph-cut Approach considering Homomorphism', IEEE Transactions on Image Processing 22,No. 12, pp. 5282-5293.

20. Xu, Y., Liang, G., Hu, G., Yang, Y., Geng, J., and Saha, P.K. (2012) 'Quantification of Coronary Arterial Stenoses in CTA using Fuzzy Distance Transform', Computerized Medical Imaging and Graphics 36.1, pp. 11-24.

21. Zreik, M., van Hamersvelt, R. W., Wolterink, J. M., Leiner, T., Viergever, M. A., and Isgum, I. (2018) 'Automatic Detection and Characterization of Coronary Artery Plaque and Stenosis using a Recurrent Convolutional Neural Network in Coronary CT Angiography', arXiv preprint arXiv:1804.04360.

22. Zreik, M., Lessmann, N., van Hamersvelt, R. W., Wolterink, J. M., Voskuil, M., Viergever, M. A. and Igum, I. (2018) 'Deep learning analysis of the myocardium in coronary CT angiography for identification of patients with functionally significant coronary artery stenosis', Medical image analysis 44, pp. 72-85.

23. Zhang, J., Gao, J., and Liu, W. (2001) 'Image Sequence Segmentation Using 3-D Structure Tensor and Curve Evolution', IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 5, pp. 629-641.