QATAR UNIVERSITY

COLLEGE OF ENGINEERING

ENCODER-DECODER ARCHITECTURE FOR ULTRASOUND IMC SEGMENTATION AND CIMT PREDICTION

BY

AISHA MORSHID AL-MOHANNADI

A Thesis Submitted to

the College of Engineering

in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Computing

June 2021

COMMITTEE PAGE

The members of the Committee approve the Thesis of

Aisha Morshid Al-Mohannadi defended on 07/04/2021.

Prof. Somaya Al-Maadeed
Thesis Supervisor

Prof. Cagatay Catal
Committee Member

Prof. Lee Yoot Khuan
Committee Member

Approved:

Khalid Kamal Naji, Dean, College of Engineering

ABSTRACT

Al-Mohannadi, Aisha, M., Masters : June: 2021, Master of Science in Computing

Title: Encoder-Decoder Architecture for Ultrasound IMC Segmentation and cIMT Prediction

Supervisor of Thesis: Prof. Somaya Al-Maadeed.

Cardiovascular diseases (CVDs) have shown a huge impact on the number of deaths in the world. Thus, Common Carotid Artery (CCA) segmentation and Intima-Media Thickness (IMT) measurement have been significantly implemented to perform early diagnosis of CVDs by analyzing the IMT feature. In this research, we aim to implement the convolutional autoencoder model to apply semantic segmentation for Intima-Media Complex (IMC) and calculate the cIMT measurement. The results were evaluated using F1 score, precision, recall, Sorenson Dice Coefficient, and Jaccard Index. We trained the encoder-decoder architecture using 80% of the dataset and 20% was left for testing. We were able to produce results of 79.92%, 74.23%, and 60.24% for the F1 Measure, Dice coefficient, and Jaccard Index, respectively. We also calculated the IMT thickness, which was 0.54mm. Our method showed that it is robust and fully automated compared to the state-of-the-art work.

DEDICATION

To my family and friends

for their continuous support

ACKNOWLEDGMENTS

I would like to sincerely thank my supervisor Prof. Somaya Al-Maadeed as well as Dr. Omar Elharrous for their guidance, support, and patience throughtout the implementation of this research. This publication was supported by Qatar University Internal Grant No. IRCC-2020-013 and Sultan Qaboos University through Grant CL/SQU-QU/ENG/20/01, respectively. The findings achieved herein are solely the responsibility of the authors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AI Artificial Intelligence.

CAE Convolutional Autoencoder.

CCA Common Carotid Artery.

CHD Coronary Heart Disease.

CNN Convolution Neural Networks.

CVAE Convolutional Variational Autoencoder.

CVD Cardiovascular Disease.

ELM Extreme Learning Machine.

FCN Fully-Connected Network.

ICA Internal Carotid Artery.

IMC Intima-Media Complex.

IMT Intima-Media Thickness.

LI Lumen-Initima.

LII Lumen-Intima Interface.

MA Media Adventitia.

MAI Media-Adventitia Interface.

ML Machine Learning.

ReLU Rectified Linear Unit.

ROI Region of Interest.

SAE Stacked Autoencoder.

SDAE Stacked Denoising Autoencoder.

SSAE Stacked Sparse Autoencoder.

SVM Support Vector Machine.

CHAPTER 1: INTRODUCTION

The heart is an essential organ in the body, where its main job is to push the blood all around the human's body. Furthermore, it is the main and central part of the cardiovascular system, which contains the blood vessels that form the blood circulation [1]. Moreover, Cardiovascular diseases (CVDs) play a great role in the worldwide death toll, and this highlights the importance of early diagnosis of such disease. According to World Health Organization (WHO), CVD is the first cause of death in the world, while taking 17.9 million lives each year [2].

According to authors in [3], CVD is an abnormal illness that affects the heart and the blood vessels, where it can be derived to the following two sub-types:

- **Coronary heart disease (CHD):** This is the most common disease that affects the heart and it involves the growth of plaques on the arterial walls.
- **Cerebrovascular disease (stroke):** It involves the formation of a barrier or interruption of blood movement to the brain.

With that being said, authors in [4] highlighted that in their study of the worldwide deaths that were caused by CVDs, almost half of the death (48.5%) was associated to coronary heart disease, while strokes only took part in 20.8% of the population tested and the rest is for other diseases. Hence, it indicates the importance of preventing the progression of CHD.

In addition, some of the risk factors of CVDs could be due to high blood pressure or high cholesterol. Hence, one of the main causes of such disease can be from a build up of inflammatory cells known as plaques, where they occur in the arterial wall resulting in blood restriction to the heart as well as lower oxygen intake. This phenomenon is known by atherosclerosis. Thus, early prediction of this disease might help in preventing the progression of atherosclerosis as well as preventing possible heart failures. In figure 1.1, it is illustrated that the plaque formation occurs mostly in the common carotid artery (CCA) and internel carotid artery (ICA).

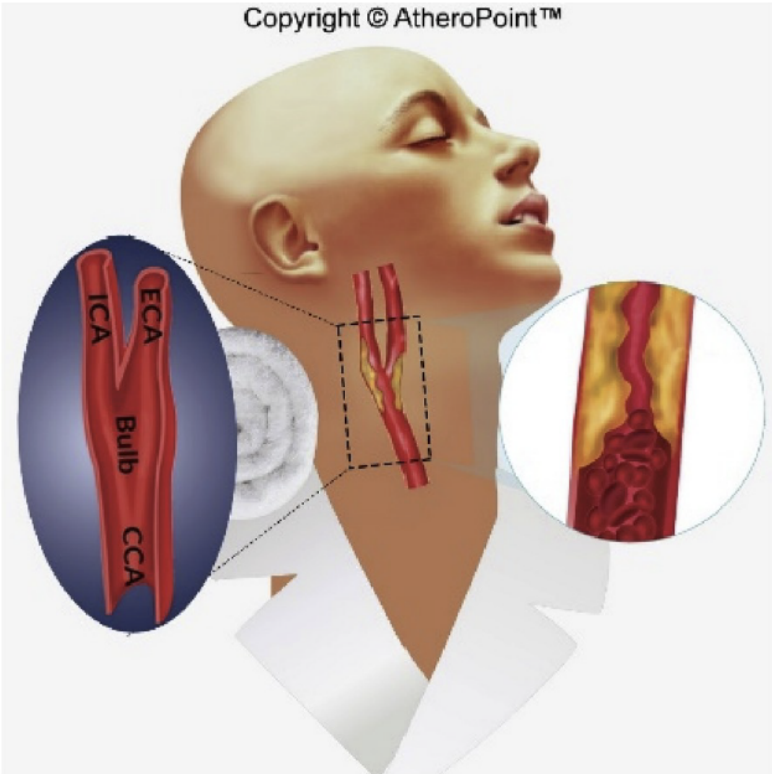

Figure 1.1: Formation of plaques in CCA [5]

One of the ways to identify the plaques in the arterial wall is by analyzing the carotid artery, as they consist of a pair of blood vessels and have several parts namely, internal, external and common parts. Plaques occur in the internal section as well as the common blood vessels of the carotid artery. Hence, plaques create a thicker wall in these vessels and it can be measured as the Intima-Media Thickness (IMT). Thus, cIMT is used as a risk marker for early prediction of heart disease, and this can be done using measurements of the difference between the lumen-initima (LI) and media adventitia (MA) walls [6]. Referring to a review done in [7], cIMT measurements showed the ability to predict CVD events independently from other risk factors, in fact, a study done by authors in [8] pointed out that it is a strong predictor for strokes even more than other vascular diseases.

Using deep learning techniques to diagnose such disease can be beneficial in many ways, namely, it can be deployed in portable devices, hence, help patients in self-diagnosing themselves. Also, it can reduce the load on doctors that might be examining and diagnosing each patient including the ones with no risks. Many applications have been conducted for cIMT segmentation and identification using deep learning and machine learning techniques, however, the accuracy of the cIMT estimation is arguable.

In this research, we focus more on evaluating the autoencoder model on IMC segmentation along with finding the best hyper-parameters for the model. This is mainly done using autoencoder networks that aim to compress the data to a latent representation and decode it using another decoder network to decompress the image, where latent representation commonly contains the features of the image. Additionally, we train and test the model using the encoder-decoder architecture as well as a dataset from [9] with pre-processing and post-processing techniques. The main aim of this research is to do segmentation of B-mode ultrasound images using deep learning encoder-decoder architecture.

1.1. Research Objectives

In this work, the main purpose is to develope a fully automated portable carotid artery screening system, which is able to segment the IMT in the arterial walls using deep learning models. Thus, deep learning models, specifically autoencoders are investigated. The main objectives of the research are summarized as the following,

- Develop Convolutional Autoencoder (CAE) model for carotid Intima-Media Complex (IMC) segmentation and IMT measurement on B-mode ultrasound images.
- Evaluate effectiveness of CAEs in variation with hyper-parameters.
- Finding an optimal architecture for CAEs by comparing effectiveness of models with state-of-the-art methods.

1.2. Research Questions

1. How does autoencoder model improve carotid IMT segmentation?
2. How is the autoencoder model unique from previous solutions?
3. Can the autoencoder with the data augmentation be effective on the given dataset with limited number of images?

1.3. Assumptions & Limitations

The main challenge and limitation of this work is the attainability of carotid IMT dataset with ultrasound images, this can limit the accuracy of the results. Nonetheless, data augmentation is expected to overcome this issue.

1.4. Thesis Structure

This thesis is organized as follows; the second chapter gives brief information on the background and concepts used in this work. Then, in the third section, a literature review on the previous work that is of relevance to the topic is discussed. The fourth chapter presents the methodology used in order to implement the solution. In chapter five and six, the results are examined, analyzed, and discussed. Finally, the seventh chapter includes the conclusion along with the future work.

CHAPTER 2: BACKGROUND AND FOUNDATIONS

In this chapter, we define concepts that are used in this research along with the dataset that is used to train the proposed model.

2.1. Convolutional Neural Networks (CNN)

In the field of Artificial Intelligence (AI) and specifically Deep Learning, many models are used and one of the popular ones is the CNN model. The architecture mainly contains an input, hidden layers, and output. Researchers in [10] defined CNNs as a deep feed-forward architecture that has the capability of generalizing compared to the other networks. They promoted that the use of CNNs give the ability of efficient object identification due to the fact that it can grasp highly abstract features. The general CNN model is explained in figure 2.1, where the main components are highlighted namely, the convolution layer, pooling layers, and fully connected layers.

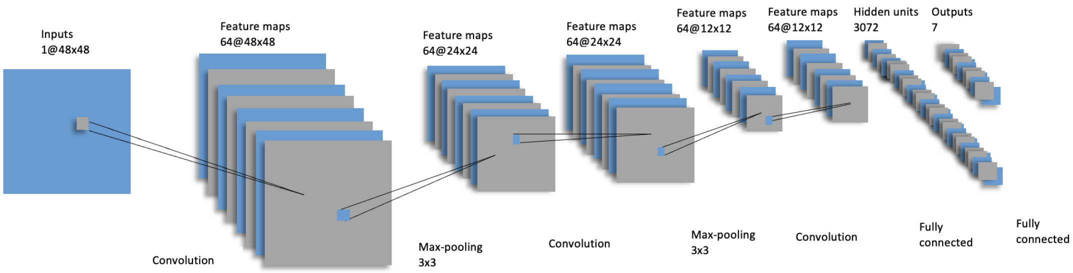

Figure 2.1: General architecture for CNNs

2.1.1. Convolution Layer

The term convolution has come to be used to refer to the mathematical operation convolution. It is illustrated in equation 2.1 that the input image X is convoluted by a kernel or certain filter.

$$Convolution = X * k \tag{2.1}$$

The convolutional operation is used in many computer vision techniques where the kernel slides over the whole image, and it illustrates how the image X is changed by the kernel. It has been stated by authors in [11] that convolution layers has the capability of extracting the feature maps using the convolution operation.

2.1.2. Pooling Layer

The pooling layer is mainly used to compress the number of trainable parameters, as this is done by the selection of a window where the input components found on that window are moved through a pooling function as shown in figure 2.2.

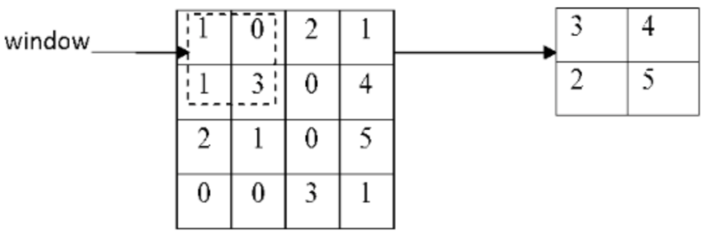

Figure 2.2: Illustration of pooling function[11]

2.1.3. Batch Normalization Layer

Batch normalization is mainly used to normalize inputs of each layer. This technique can help in avoiding the over-fitting problem, which occurs during the training session. Thus, their steps during the training are to calculate the mean and variance of the input layer and normalize it using calculations of the batch statistics done previously. Then, the final output is found by scaling and shifting.

2.1.4. Rectified Linear Unit (ReLU)

ReLU is an activation function, where it is widely used in neural networks, specifically CNNs. The main functionality of it is to output the same value if it is positive, or give zero if the value is negative. It can be represented as follows:

$$f(x) = max(0, x) \tag{2.2}$$

One if its advantages is when training as it is not time consuming since it does not have complicated mathematics. However, it may cause problems when the output is always zero, which may end up with dead neurons.

Parametric ReLU (PReLU)

It is one type of ReLU, however, instead of giving an output of zero, it gives a parametric value which is learned during the training session and its equation can be represented as follows:

$$f(x) = ax \tag{2.3}$$

There is another type of ReLU, which is Leaky ReLU and it has a predetermined value of a = 0.01. On the other hand, PReLU gives more optimized a value where it can be learnt to find the best a value for the function. This technique overcomes the problem of dead neurons in the ReLU function and it helps in speeding the training session.

2.1.5. Fully Connected Layers

This layer is done in the final stage of the model, where it contains the final output of the convolution and pooling layers. Moreover, it is calculated using the dot product of the final input along with the weights to produce the final output.

2.2. Autoencoders

The term autoencoders has been applied to refer to the encoder-decoder model. They were defined by authors in [12] as a neural network that accordingly determine effective features and representations from the data. Hence, it simplifies the process of feature engineering in addition to compressing dimensionality. The main architecture for autoencoders is described in figure 2.3, where it consists of an input that is encoded and compressed to a representation of a code, and an output which is produced using decompression of the coded features.

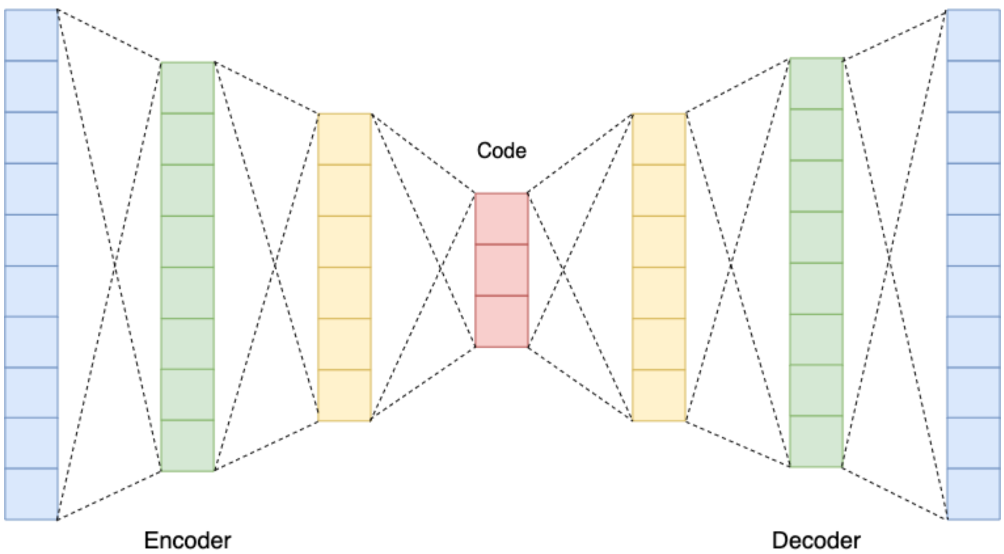

Figure 2.3: General architecture for autoencoders

Additionally, autoencoders can have a variety of types which were explained by authors in [13] and illustrated in figure 2.4. The variational autoencoders are mostly used when desiring to manage latent distribution, whereas sparse autoencoders have larger hidden nodes than input nodes, they can still locate significant features from the data. They contain sparsity penalty to avoid overfitting.

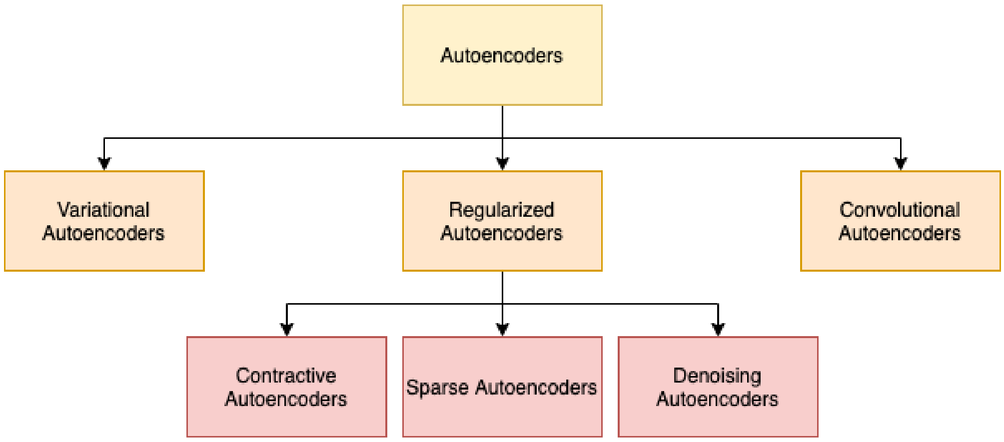

Figure 2.4: Types of autoencoders

Furthermore, there is the denoising autoencoder, as it has the ability the eliminate noise coming from the input and produce a clear output. Unlike denoising autoencoders, contractive autoencoders has powerful learned representations with noise removal as well. There is also convolutional autoencoders which is explained in details in section 2.2.1.

2.2.1. Convolutional Autoencoders

The convolutional autoencoders (CAE) consist of convolutional layers in both the encoder and decoder parts. As described in section 2.1, convolutional layers are used to calculate the change in an image. In CAE encoder, the maxpooling and convolution layers are used to apply feature extraction on images. Then, it produces a reduced size features of the input and represent it as code. The decoder then takes the code and decodes it using convolution and upsampling layers to reconstruct the image [14].

The figure 2.5 shows the SegNet model, which is a model used by convolutional autoencoders without fully connected layers. This model is used mainly for semantic segmentation. The image was inspired by authors in [15].

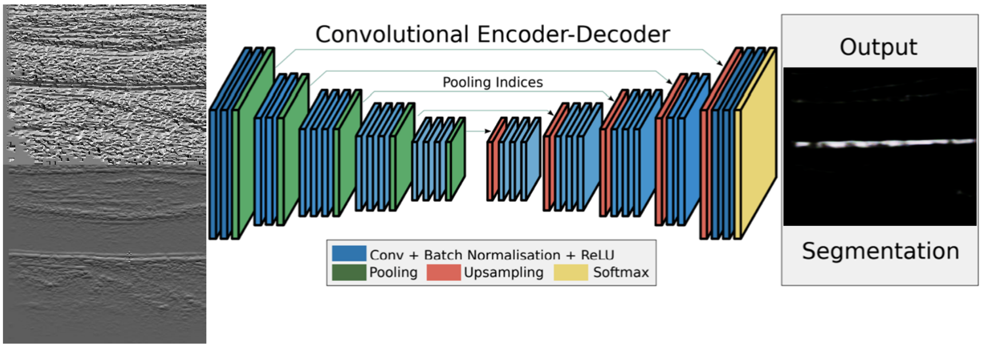

Figure 2.5: The general architecture for SegNet

2.3. Dataset

The dataset used for this thesis is a dataset in [9]. It contains 100 carotid IMT B-Mode ultrasound images with their ground truth points determined by two clinical experts. In their work Loizou et al. [9], highlighted that images were taken from 42 female and 58 male symptomatic patients aging between 26 and 95, where they produced longitudinal ultrasound images.

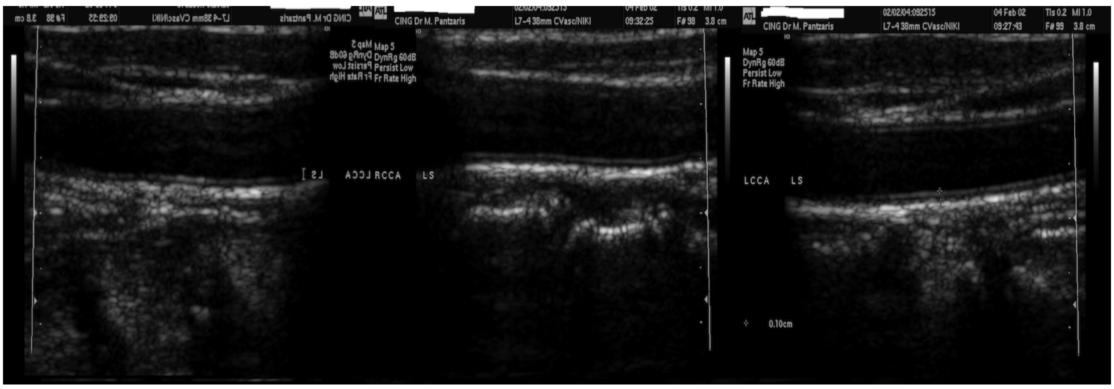

Figure 2.6: Three sample images from the dataset [9]

The figure in 2.6 illustrates 3 sample images from the dataset that we work with in this thesis. Each sample has its own frame and these frames were excluded from the images as pre-processing step since we only need the ultrasound image.

Similarly, authors in [9] identified the IMT measurements from both experts with number of techniques. They used speckle reduction, as well as normalization as pre-processing steps. In this research, we focus only on normalized images, hence, only IMT measurements for normalized images are used. The table 2.1 shows the experts measurements for IMT in the case of normalized images.

Table 2.1: Ground truth measurements for IMT

	At time 0 months	At time 12 months
Expert	mean IMT measurement (mm)	mean IMT measurement (mm)
1	0.68	0.68
2	0.61	0.57

The experts readings included two time periods one at time 0 months and the other at time 12 months. According to others in [9], this was done to test the intra-observer variability for the same expert. This means that the experts highlighted the carotid walls two times in different period of time for the same image in order to assess observer errors.

CHAPTER 3: LITERATURE REVIEW

This chapter introduces the previous methods done for carotid IMT segmentation, as well as medical applications using autoencoders, where there is a considerable amount of literature on carotid artery IMT segmentation using deep learning, machine learning, and contour techniques. The sections is structured as follows; The autoencoder application in medical field are introduced briefly. Then, the segmentation techniques for cIMT is tackled, followed by identification of the shortcomings of the current literature that has been implemented regarding this work.

3.1. Medical Autoencoder Applications

Currently, autoencoders are growing in the medical imaging field, as they showed that they can have many types including merged techniques such as stacked autoencoders (SAE), stacked denoising autoencoders (SDAE), stacked sparse autoencoders (SSAE), and convolutional variational autoencoders (CVAE). Many studies have been published on CVAE including medical application to predict post-trauma health outcomes [16]. Another study was done by authors in [17], which included using CVAE to automatically detect plant diseases, as well as authors in [18] developed CVAE based system for Electrocardiopraphic imaging (ECGI).

Several studies have been conducted for autoencoders in medical applications namely, mortality risk prediction [19], as well as chest radiology improvement using denoising autoencoders [20]. Regarding image segmentation in medical field, autoencoders show huge impact on the accuracy of applications comparing to other models. Thus, authors in [21] claimed that their application for 3D image segementation using CT scans show improvement in results.

3.2. Carotid IMT Segmentation Applications

Many attempts have been done regarding carotid IMT segmentation and classification, however, only few that show competitive results given the fact that IMT segmentation is the most sensitive step as the thickness measurements depends on the accuracy of the IMT segmentation. Authors in [22] use Support Vector Machines in order to train and segment the carotid IMT. In their method, they used 49 ultrasound images and divided them into two sets with 50% for training and the rest for testing. Their method provided 93% accuracy, and as for the IMT measurement they found it to be 0.66mm.

One of the first attempts for carotid segmentation was done by Loizou et al. (2013) [23], where they implemented a semi-automated snake's-based segmentation system proper for complete CCA segmentation. Their method concentrated at estimating IMT measurements from manually defining the carotid plaque and diameter and then applying the snake's algorithm to get the measurements. The dataset that was used for this algorithm was 300 2D ultrasound images. Their method did not result in significant difference from the state-of-the-art, as it was limited to only manual readings.

Improving on previous work authors in [24] attempted to implement a fully automated segmentation system using adaptive snake's contour as well as level set segmentation. When comparing both techniques together authors found out that the snake's contour method outperformed the level set segmentation. Another technique was developed by authors in [25] that does not depend on AI as well, their method included bulb edge detection and then segmental IMT measures are applied according to the edge detected. Their dataset consisted of 649 images that has between moderate and heavy lighting. They got a significantly low error in calculating the IMT measurement which is around 0.0106mm and Precision of Merit that equals to 98.23%.

Another technique was built by authors in [26] that avoided the implementation of deep learning for IMT segmentation. The authors illustrated that they used wind driven optimization technique for carotid IMT segmentation, as they focused on developing a fully automated Region of Interest (ROI) extraction as well as they used for intima media complex a threshold-based method. Their results included IMT measurement of 0.69mm as they claimed that their method outperformed other work in the literature.

Experiments on IMT segmentation was not limited to non-AI only, where authors in [5] implemented a screening tool that integrates a two-stage artificial intelligence model for IMT and carotid plaque measurements, which consists of CNN and Fully Convolutional Network (FCN). The system goes through two deep learning models, as the first divides the CCA from the ultrasound images into two categories the rectangular wall and non-wall patches. Then, the region of interest is analyzed and fed to the second stage, where they identify some features to calculate the carotid IMT and the plaque total as well. Furthermore, their dataset consisted of 250 images, whereas their results while using the proposed AI model showed an error of IMT measurement that equals to 0.0935mm.

As investigations of IMT segmentation went on with deep learning and machine learning, authors in [27] proposed a method for segmentation using CNN. Therefore, the researchers applied an algorithm that finds the ROI using the CNN architecture which includes 8 layers. Moreover, they trained the network using 220 left and right CCA images for ROI localization. After that, the intima media complex area is extracted in order to measure the IMT. The mean difference for IMT measurement is found to be 0.08mm, where they got the accuracy to be equal to 89.99% for the CNN network.

Another research group [28] investigated IMT segmentation in video interpretation of IMT measurement using CNN. They performed CNN using 6 layers, and they were able to achieve low error rate in their measurements and they got a result of 2.1mm error with only one failure for testing subjects. Furthermore, another technique was used by Joseph and Sivaprakasam (2020) [29], where they used double line echo patterns coming from the B-mode and A-mode ultrasound images to identify both arterial walls. Their method showed an error of IMT measurement that equals to 0.18mm.

Another combined method was implemented by researchers in [30], where they used deep learning for IMT measurement for patients with diabetes. Their method includes two stages, the first is CNN network that is used for segmentation and the other is machine learning based regression. Therefore, their output was the borders of the lumen intima and the media-adventitia which is used to calculate the carotid IMT. In their work, they used a dataset of 396 B-mode ultrasound images, as they got the result of the error for cIMT measurement to be around 0.126mm. Researchers claimed that their method was 20% improved comparing to other non-deep learning methods.

One more deep learning method was discussed by researchers in [31], where they used CNN with multiple hidden layers for image classification. They were able to test the network using 501 ultrasound images dataset and achieve an accuracy of 89.1% for IMT classification. The other method was developed by authors in [32], where they used four classification algorithms for IMT measurement, the algorithms consisted of SVM with linear kernel, SVM with radial basis kernel, AdaBoost, and random forest. They evaluated their method using a dataset that consisted of 29 images, and they concluded that the best results were for integrated random forest method which results in 80.4% sensitivity and 96.5% specificity.

One study has been made regarding IMT measurements using autoencoders and this is done by authors in [33], their method included ROI prediction and then Lumen-Intima Interface (LII) and Media-Adventitia Interface (MAI) walls predictions in the predicted ROI, as their dataset consisted of 67 images. Authors claimed that they used Extreme Learning Machines (ELM) along with autoencoders in order to distinguish which block is included in the ROI and which is not. Whereas the LII and MAI recognition was done using pixel classification. Moreover, they evaluated their IMC segmentation by using accuracy, specificity, sensitivity, and Matthews correlation coefficient (MCC). Their results used sensitivity and specificity for evaluating ROI prediction, on the other hand, accuracy and MCC were used LII and MAI. Final results were mean IMT measurement of 0.625 $\pm$ 0.1673mm, with accuracy for LII as 99.30% and MAI as 98.8% and MCC for LII and MAI as 98.03% and 97.05%, respectively.

Given the above methods, one research used autoencoders for IMT measurement which is the one done in [33]. Their findings were done using machine learning and autoencoders for ROI localization only, where they used another technique for IMT segmentation and recognition. Moreover, some limitations were identified such as, using semi-automated systems could lower the feature of having a portable system, also some methods used clinical instruments that are not portable. Furthermore, when compare to non-AI methods, we observe that error found can be lower and accuracy can be enhanced further when using AI methods. To illustrate more, we conclude that work done in [30] has lower error calculated in IMT measurement (0.126mm) than the one in [29], which is 0.18mm. Therefore, in our research we focus on implementing a solution that is fully automated and supports portability along with taking into account segmentation metrics. Also, a summary of the literature can be found in table. 3.1. The table shows that the highest accuracy is [33], however, their

One more thing to point out is that table 3.1 shows the different applications that has been done for IMT predications including AI and non-AI techniques. However, comparing these applications together may not be fair since each application uses different set of dataset as well as the percentage of the dataset for training that was used is not the same. With that being said, we can make relative comparisons where we can point out the outcomes of each application given their architecture or method used. Thus, in this work, we did not compare our results quantitatively with the provided literature since different datasets are used. Instead, comparison is only done with application [9].

Table 3.1: Summary for the literature review

Ref	Classifier	Evaluation method	IMT measurement (mm)	No. images	Other evaluation methods	AI?
[22]	SVM	The correlation coefficient R	GT = 0.67, IMT = 0.66	49	accuracy: 93%	Yes

Ref	Classifier	Evaluation method	IMT measurement (mm)	No. images	Other evaluation methods	AI?
[23]	snake's segmentation	Wilcoxon-sum test	GT= sympto: 0.96, normal: 0.75, IMT= sympto: 0.95, normal: 0.74	300	-	No
[24]	snake's contour, level set segmentation	Wilcoxon-sum test	error in IMT SC: 0.12, LS: 0.09	100	-	No

Ref	Classifier	Evaluation method	IMT measurement (mm)	No. images	Other evaluation methods	AI?
[25]	bulb edge detection	bulb closeness factor, IMT measurement	error in IMT: 0.01603 ± 0.0031	Dataset 1: 172, Dataset 2: 649	precision accuracy: 98.23%, sensitivity: 98.6%, specificity: 93.9%	No
[26]	wind driven optimization technique	The correlation coefficient R	GT= 0.704±0.216	Dataset 1: 100, Dataset 2: 25	-	No
[5]	CNN, FCN	correlation coefficient, Polyline distance metric (PDM)	error in IMT: 0.0935±0.0637	250	accuracy for characterization: 99%	Yes

Ref	Classifier	Evaluation method	IMT measurement (mm)	No. images	Other evaluation methods	AI?
[27]	CNN	accuracy	difference of IMT measurement: 0.08	220	accuracy: 89.99%	Yes
[28]	CNN	-	error for bulb localization: 2.1	92 videos	-	Yes
[29]	Signal processing	accuracy of RF frame sequqences with different SNR	mean absolute error for IMT: 0.18	40	-	No
[30]	FCN, CNN, regression	PDM, Precision of Merit (PoM)	error in IMT: 0.126±0.134	396	-	Yes

Ref	Classifier	Evaluation method	IMT measurement (mm)	No. images	Other evaluation methods	AI?
[31]	CNN	Precision, recall, f1score, support	-	501	accuracy:89.1%, sensitivity:89%, specificity: 88%	Yes
[32]	SVM	specificity, sensitivity, dice coefficient	-	29	sensitivity: 80.4%, specificity: 96%, dice coeff: 81%	Yes
[33]	ELM autoencoder	accuracy, specificity, sensitivity, Matthews correlation coefficient	mean absolute different: 0.625±0.1673	67	accuracy LII:99.30% MAI:98.80%, MCC LII:98.03% MAI:97.05%	Yes

CHAPTER 4: METHODOLOGY

In this chapter, we describe the procedure used in order to implement our solution. The steps include, model architecture design, instrumentation, and implementation stages.

4.1. Introduction

It has been discussed that autoencoders have a variety of types, thus, in this work we focus on the convolutional autoencoders, specifically SegNet model, and in this section we go in details into SegNet model and its architecture.

4.1.1. SegNet

As described in section 2.2.1, convolutional autoencoders can be implemented for segmentation as SegNet model. It mainly contains convolution, pooling, and batch normalization layers as shown in Figure 2.5. Recently, it has been used to implement medical applications such as the work done in [34]. Initially, it has been developed by Vijay Badrinarayanan et al. [35], as it is mainly used for outdoor images or indoor images containing many classes. The most unique feature in SegNet model is that it does not have fully connected layers, which reduces the number of parameters. Also, the encoder part contains downsampling layers with convolutional layers and it is the opposite for decoder, where it contains upsampling layers along with convolutional layers.

4.2. Model Architecture

In our proposed solution, we use convolutional autoencoder to train two inputs where each input has certain pre-processing technique that aids the model to distinguish IMT better. The model consists of SegNet architecture. The first part in the architecture is the encoder, where it uses convolutional layers with 3x3 kernels, batch normalization and padding along with ReLU layers and pooling layers. Firstly, the two inputs are encoded separately using the same convolutional architecture, then they are concatenated to produce one encoder to be fed to the decoder model.

As for the decoder part, it does the upsampling and decoding of the encoded layers. The convolutional layer in this section is used for strengthening the sparse feature maps created by the encoder by transforming it into final labels. The final SegNet layer calculates the multinomial loss with the SoftMax layer. Hence, the final architecture is shown in Figure 4.1.

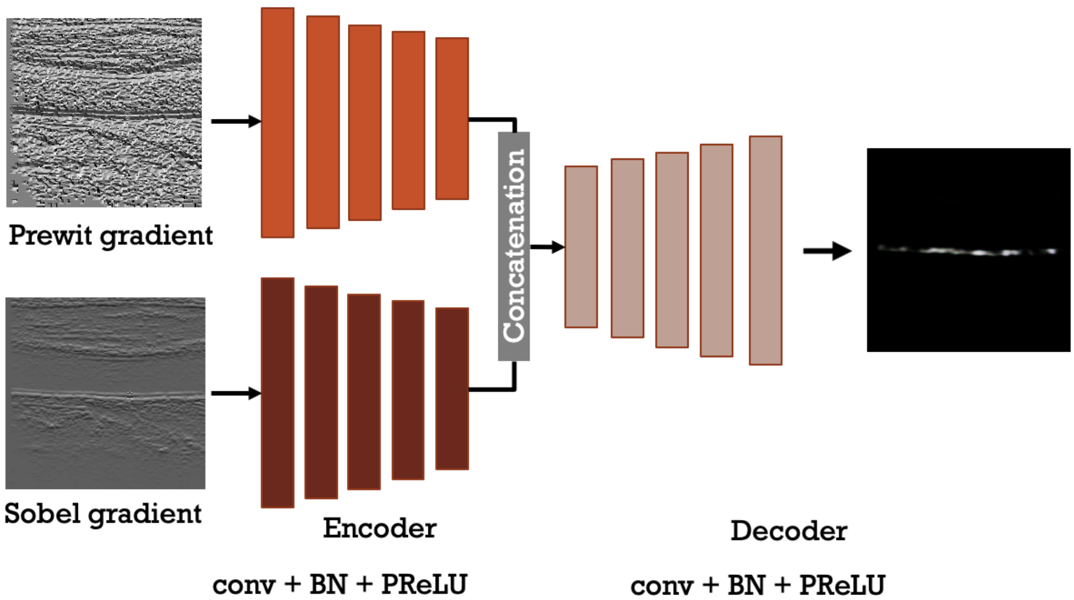

Figure 4.1: The model architecture used for the research solution

For this thesis, we use two components namely, sobel gradient direction image and prewitt gradient direction image as inputs of the model. Also, these images went through pre-processing steps, which is discussed in section 4.5.

4.3. Instrumentation

In order to implement the architecture and train it using the model described in section 4.2, we use two programming languages, the first is MATLAB and second is python, as the reason behind using them is due to the fact that MATLAB contains many functions for pre-processing and postprocessing in image processing, while python is suitable for training the proposed model, evaluating it, and testing it.

Table 4.1: Environment Setup (Python)

Operating System	Windows 10
GPU	NVIDIA Geforce
Framework	Anaconda
Languages	Python 2.7

Table 4.2: Environment Setup (MATLAB)

Operating System	MacOS 11.2.2
GPU	-
CPU	2.4 GHz Quad-Core Intel Core i5
Framework	MATLAB

4.4. Performance Evaluation

The evaluation of the deep learning model performance computed in the testing phase was based on the following main evaluation metrics:

- **Precision**: This calculates how close the values are from each other and how close they are from the true values.

- **Recall**: also known as sensitivity. This is the ratio of the correct results by the overall correct data.

- **F1 Measure**: This is calculated using both precision and recall, where it gives an overall overview of the performance of the system.

$$F1\ Measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4.1}$$

- **Sorensen Dice Coefficient**: This calculates the similarity of two samples and mainly used to validate image segmentation algorithms. It is also more about the percentage of overlap between two images.

$$Dice = \frac{2 * TP}{2 * TP + FP + FN} \tag{4.2}$$

- **Jaccard Index**: This is the percentage of similarity for two images. It is similar to Dice Index, however, the jaccard index takes into account true positive only once, while in the Dice Coefficient it does it twice.

$$Jaccard\ Index = \frac{TP}{TP + FP + FN} \tag{4.3}$$

We focus mainly about the F1 measure, Dice coefficient, and the Jaccard index in this work, as they mainly evaluate the similarity and the efficiency of the model and segmentation algorithm.

4.5. Implementation

In this section, we describe the stages that we went through in order to implement our solution. The main stages are, pre-processing, data augmentation, segmentation, and post-processing.

4.5.1. Pre-processing

First of all, we took the raw images and removed the frames that were not of interest. After that, images were normalized and taken to be processed using Sobel and Prewitt gradient methods that are available built-in functions in MATLAB. Both methods produced gradient magnitude as well as gradient direction. For this implementation, we stored only normalized gradient direction for both methods. After trying other filters, the gradient directional images were found to be the most accurate that shows the cIMT clearly than the rest of the images.

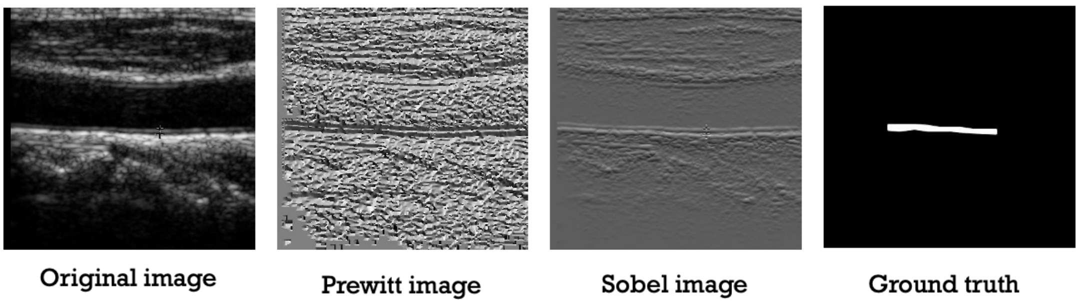

Figure 4.2: The pre-processing phase

As for the ground truth points, we converted the points to binary images in order to input them as labels in the deep learning model, as well as to compare them with predicted images in the testing phase. In Figure 4.2, the original image along with the gradient images and the produced ground truth image are shown. Additionally, in order to train the model with data augmentation, the new generated ground truth mask images were produced using lines that connects the ground truth points given in the dataset.

4.5.2. Data Augmentation

Data augmentation is mainly used when we have small dataset and would like to increase the number of images in a given dataset. Thus, it provides small operations that can give the ability to rotate, flip, shift, zoom, or translate a given image without changing the content of it. Hence, we keep the final image as the original image features. Moreover, in order to do data augmentation, we need to have binary mask images since we are changing the display of an image, then the given mask should go through the same process.

In this stage, we use special features to implement augmentation namely, rotation, width and height shift, and zoom. Table 4.3 shows the values used for the augmentation. The augmentation was done using ImageDataGenerator library in python, where it was used to augment both image and its binary mask.

Table 4.3: Data augmentation parameters

Feature	Value
Rotation	10
Width shift	0.2
Height shift	0.2
Zoom	0.2

4.5.3. Segmentation

In the implementation of the model, we used 80% of the final dataset and converted it to both sobel gradient and prewitt gradient. These images were fed to the encoder-decoder architecture described in Figure 4.1. During the training phase, we trained the model multiple times in order to get the best performance and tune the hyper-parameters for better accuracy. Finally, the training was done using 50 epochs along with 10 steps per epoch, which means it increases the data augmentation for each epoch 10 times.

4.5.4. Post-processing

The final segmented image had some noise that needed to be reduced and specific Region of Interest (ROI) needed to be highlighted. For that, we used morphological opening, which removes any small noise in the image and can detect discontinued blobs. Also, we used morphological closing in order to avoid having a discontinued segmentation of the carotid artery IMT.

CHAPTER 5: RESULTS

During the implementation of the deep learning model, we did extensive training as we ran the experiments various of times with different number of epochs. This included changing the batch size as well as experimenting with the input images along with the pre-processing techniques.

Firstly, the training was done on the prewitt and sobel images as inputs with batch size equals to 32 and 8 as well as we included data augmentation in this phase. As a result, it was clear that the 32 batch size was not segmenting only the desired part and 8 batch size was more accurate. Thus, we trained the model using 8 batch size and augmentation in the second phase. Furthermore, we examined with changing input images as prewitt and sobel, and the other experiment we made the inputs as original image and sobel image. However, the two gradient images were giving more accurate results. The first results were done using the architecture without batch normalization layers. We examined then the batch normalization layers on the final results and it gave preferable results than the outputs previously examined.

Similarly, table 5.1 illustrates the trials that were done and how the parameters were changed. The last two trials included the batch normalization layer, which had the higher percentages in the final results explained in table 5.3 and 5.4. Also, figures A.1, A.2, and A.3, show samples of the results from the trials. In addition to table 5.2, where the performance of each trial is provided.

Table 5.1: Experimental Trials

Trial	Input 1	Input 2	Batch size	Epochs	steps/epoch	Learning rate	Augmentation
1	Prewitt	Sobel	32	50	-	0.0001	No
2	Prewitt	Sobel	8	50	-	0.0001	No
3	Prewitt	Sobel	8	15	5	0.0001	Yes
4	Original	Sobel	8	5	50	0.0001	Yes
5	Prewitt	Sobel	8	7	80	0.00001	Yes
6	**Prewitt**	**Sobel**	**8**	**50**	**10**	**0.00001**	**Yes**

Table 5.2: Experimental Trials

Trial	Input 1	Input 2	Batch size	F1 Measure	Jaccard Index	Dice Coefficient
1	Prewitt	Sobel	32	-	50.77%	36.93%
2	Prewitt	Sobel	8	64.92%	45.65%	60.44%
3	Prewitt	Sobel	8	70.77%	45.43%	60.51%
4	Original	Sobel	8	67.63%	46.64%	61.31%
5	Prewitt	Sobel	8	73.63%	52.29%	66.07%
6	**Prewitt**	**Sobel**	**8**	**79.92%**	**60.24%**	**74.23%**

After tuning the hyper parameters and decreasing the learning rate we were able to get the results shown in figure 5.1.

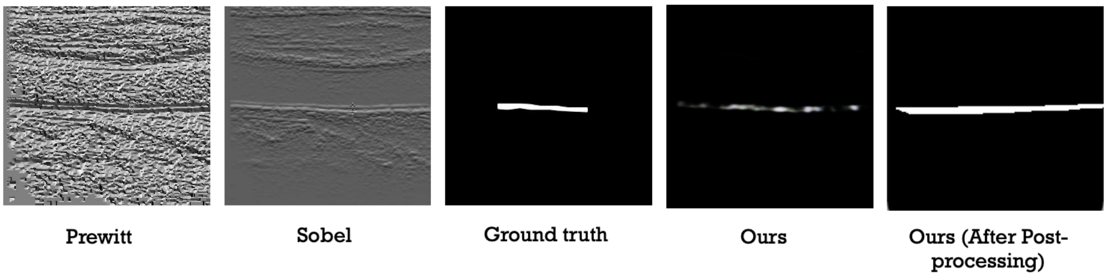

Figure 5.1: The results of one image for the final training

In addition, the results show better similarity with the ground truth with little extension of the line. During the testing phase, we evaluate the model using three metrics discussed in section 4.4. The results of the metrics are showed in table 5.3.

Table 5.3: Initial Model Architecture Results for Trial 5 in Table 5.1

Metric	Autoencoder model
F1 Measure	73.63%
Precision	77.33%
Recall	78.92%
Dice Coefficient	66.07%
Jaccard Index	52.29%

Table 5.4: Improved Model Architecture Results

Metric	Autoencoder model
F1 Measure	79.92%
Precision	81.18%
Recall	82.06%
Dice Coefficient	74.23%
Jaccard Index	60.24%

According to the jaccard index and the dice coefficient, they show a similarity of the tested data with the binary masks. The highest percentage is the F1 measure, where it gives an overview of the performance of the system.

The results showed that the system has somewhat good performance, however, it can be further enhanced, where pre-processing or post-processing techniques need to be further enhanced. Results might not give the best accuracy due to the fact that the dataset is not very clean, as it was hard to work with it.

Moreover, we perform pixel calculations to get the thickness of the predicted IMT measurement. The calculations were made by calculating the distance from the upper boundary to the lower boundary. It was done using MATLAB functions bwdist(), as it calculates the vertical distance of binary object. Also, local max value was taken and then, the mean value was calculated for all tested images to get the thickness as

2.989 pixels. Furthermore, converting pixels to mm we get 0.54mm as the mean IMT measurement.

In comparison to the work done in [9], as well as the ground truth, the table 5.5 illustrates the error found in both the dataset and the proposed method compared to the ground truth. As explained before in table 2.1, the ground truth IMT has been determined by two experts in certain time.

Table 5.5: Comparison of the results

	Expert 1	Expert 2	snake's segmentation [9]	Proposed solution
Normalized mean IMT measurement(mm)	at time 0,12: 0.68	at time 0,12: 0.61, 0.57	0.67	0.54
Error in IMT	-	-	Expert 1: at time 0,12: **0.01**, Expert 2: at time 0,12: **0.06,0.1**	Expert 1: at time 0,12: **0.14**, Expert 2: at time 0,12: **0.07,0.03**

CHAPTER 6: DISCUSSION

Given the results discussed in section 5, we were able to achieve a segmented region for the carotid IMT, which was then used to estimate the thickness. For this thesis, we were able to train and test the images and compare them to the ground truth points.

During the implementation of this solution, many other architectures were investigated including, UNet segmentation using MATLAB. These models were trained using more than 50 epochs with no good results. Therefore, the encoder-decoder architecture was able to produce segmented output which achieved a good performance. The results of this model look promising and are good for future expansion.

One of the main challenges faced in this research is finding a good segmented and annotated dataset. We faced many issues to get a dataset and we were able to receive the dataset that we worked on. Moreover, the dataset was not clean enough to be processed, hence, it was time consuming to work on these images, where in some cases the IMT was not very clear. Thus, the output for these images from the model are discontinued parts of sections around the IMT. Also, the dataset has no recent studies which makes it hard to compare between results.

According to the research questions described in section 1.2, we can conclude now that the model was able to segment IMC fairly. However, due to the lack of variety of images in the given dataset, it is not clear if the model can improve IMT segmentation. Thus, further research needs to be done regarding the dataset. Moving to the second question, the model chosen has not been used before for IMT segmentation and it has shown good results for this dataset and it is open for further improvements. We also observed from the trials and experimentation that 8 batch output with augmentation showed better segmentation than the one without augmentation. Hence, the data augmentation was effective on the dataset along with the autoencoder model.

In general, after comparing with the results found in [9], we identify that the proposed method is robust and fast and is fully automated compared to their semi automated snake segmentation.

CHAPTER 7: CONCLUSION

In conclusion, CVDs take millions of lives on yearly basis, which means it is important to provide people with ways to early diagnosis such disease. Many implementations were done for such problem using computer vision techniques for B-mode ultrasound images. Furthermore, we studied recent work in the field of carotid intima media thickness segmentation as well as autoencoder applications. In this thesis, we were able to study a deep learning model, specifically convolutional autoencoder and find the best hyper-parameters and architecture, which provided similar results to the given ground truth in the dataset. We trained the autoencoder architecture using 10 steps per epoch and 50 epochs and 80% of the dataset. We were able to obtain results of 79.92%, 74.23%, and 60.24% for the F1 Measure, Dice coefficient, and Jaccard Index, respectively. We also calculated the IMT thickness, which was 0.54mm. The model showed good performance with a lowest error 0.03mm compared to the ground truth data.

Further enhancement could be done by experimenting with the optimized model along with other ultrasound B-mode carotid datasets, this would give an overview of the generality of such system and the performance given other images. Our proposed system is highly recommended to be used along in a portable device that acquire ultrasound images and process it in order to give patients the ability to early diagnose themselves.

REFERENCES

[1] L. Yahaya, N. David Oye, and E. Joshua Garba, “A comprehensive review on heart disease prediction using data mining and machine learning techniques,” *American Journal of Artificial Intelligence*, vol. 4, no. 1, p. 20, 2020. DOI: 10.11648/j.ajai.20200401.12.

[2] *Cardiovascular diseases*. [Online]. Available: https://www.who.int/health-topics/cardiovascular-diseases/.

[3] *Coronary heart disease*, Jun. 2010. [Online]. Available: https://www.healthknowledge.org.uk/public-health-textbook/disease-causation-diagnostic/2b-epidemiology-diseases-phs/chronic-diseases/coronary-heart-disease.

[4] M. Hamer, G. O’Donovan, and E. Stamatakis, “Association between physical activity and sub-types of cardiovascular disease death causes in a general population cohort,” *European Journal of Epidemiology*, vol. 34, no. 5, pp. 483–487, 2018. DOI: 10.1007/s10654-018-0460-2.

[5] M. Biswas, L. Saba, S. Chakrabartty, N. N. Khanna, H. Song, H. S. Suri, P. P. Sfikakis, S. Mavrogeni, K. Viskovic, J. R. Laird, E. Cuadrado-Godia, A. Nicolaides, A. Sharma, V. Viswanathan, A. Protogerou, G. Kitas, G. Pareek, M. Miner, and J. S. Suri, “Two-stage artificial intelligence model for jointly measurement of atherosclerotic wall thickness and plaque burden in carotid ultrasound: A screening tool for cardiovascular/stroke risk assessment,” *Computers in biology and medicine*, vol. 123, p. 103 847, Aug. 2020, ISSN: 0010-4825.

DOI: 10.1016/j.compbiomed.2020.103847. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2020.103847.

[6] F. Molinari, K. M. Meiburger, L. Saba, U. Rajendra Acharya, M. Ledda, A. Nicolaides, and J. S. Suri, "Constrained snake vs. conventional snake for carotid ultrasound automated imt measurements on multi-center data sets," *Ultrasonics*, vol. 52, no. 7, pp. 949–961, 2012. DOI: 10.1016/j.ultras.2012.03.005.

[7] A. Ravani, J. Werba, B. Frigerio, D. Sansaro, M. Amato, E. Tremoli, and D. Baldassarre, "Assessment and relevance of carotid intima-media thickness (c-imt) in primary and secondary cardiovascular prevention," *Current Pharmaceutical Design*, vol. 21, no. 9, pp. 1164–1171, 2015. DOI: 10.2174/1381612820666141013121545.

[8] Y. Saxena, V. Saxena, M. Mittal, M. Srivastava, and S. Raghuvanshi, "Agewise association of carotid intima media thickness in ischemic stroke," *Annals of Neurosciences*, vol. 24, no. 1, pp. 5–11, 2017. DOI: 10.1159/000464417.

[9] S. Petroudi, C. Loizou, M. Pantziaris, and C. Pattichis, "Segmentation of the common carotid intima-media complex in ultrasound images using active contours," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 11, pp. 3060–3069, 2012. DOI: 10.1109/TBME.2012.2214387.

[10] S. Indolia, A. K. Goswami, S. Mishra, and P. Asopa, "Conceptual understanding of convolutional neural network- a deep learning approach," *Procedia Computer Science*, vol. 132, pp. 679–688, 2018, International Conference on Computational Intelligence and Data Science, ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2018.05.069.

[11] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018, ISSN: 0031-3203. DOI: `https://doi.org/10.1016/j.patcog.2017.10.013`.

[12] W. H. Lopez Pinaya, S. Vieira, R. Garcia-Dias, and A. Mechelli, "Chapter 11 - autoencoders," in *Machine Learning*, A. Mechelli and S. Vieira, Eds., Academic Press, 2020, pp. 193–208, ISBN: 978-0-12-815739-8. DOI: `https://doi.org/10.1016/B978-0-12-815739-8.00011-0`.

[13] D. Bank, N. Koenigstein, and R. Giryes, *Autoencoders*, 2020. arXiv: `2003.05991 [cs.LG]`.

[14] E. Blanco-Mallo, B. Remeseiro, V. Bolón-Canedo, and A. Alonso-Betanzos, *On the effectiveness of convolutional autoencoders on image-based personalized recommender systems*, 2020. arXiv: `2003.06205 [cs.LG]`.

[15] V. Badrinarayanan, A. Kendall, and R. Cipolla, *Segnet: A deep convolutional encoder-decoder architecture for image segmentation*, 2016. arXiv: `1511.00561 [cs.CV]`.

[16] A. S. Cakmak, N. Thigpen, G. Honke, E. P. Alday, A. B. Rad, R. Adaimi, C. J. Chang, Q. Li, P. Gupta, T. Neylan, S. A. McLean, and G. D. Clifford, *Using convolutional variational autoencoders to predict post-trauma health outcomes from actigraphy data*, 2020. arXiv: `2011.07406 [cs.LG]`.

[17] V. Zilvan, A. Ramdan, E. Suryawati, R. Kusumo, D. Krisnandi, and H. Pardede, "Denoising convolutional variational autoencoders-based feature learning

for automatic detection of plant diseases," Oct. 2019, pp. 1–6. DOI: 10.1109/ICICoS48119.2019.8982494.

[18] T. Bacoyannis, J. Krebs, N. Cedilnik, H. Cochet, and M. Sermesant, "Deep learning formulation of ecgi for data-driven integration of spatiotemporal correlations and imaging information," in. May 2019, pp. 20–28, ISBN: 978-3-030-21948-2. DOI: 10.1007/978-3-030-21949-9_3.

[19] Z. Alhassan, D. Budgen, R. Alshammari, T. Daghstani, A. S. McGough, and N. Al Moubayed, "Stacked denoising autoencoders for mortality risk prediction using imbalanced clinical data," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2018, pp. 541–546. DOI: 10.1109/ICMLA.2018.00087.

[20] D. Lee, S. Choi, and H.-J. Kim, "Performance evaluation of image denoising developed using convolutional denoising autoencoders in chest radiography," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 884, pp. 97–104, 2018, ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2017.12.050.

[21] C. Sital, T. Brosch, D. Tio, A. Raaijmakers, and J. Weese, *3d medical image segmentation with labeled and unlabeled data using autoencoders at the example of liver segmentation in ct images*, 2020. arXiv: 2003.07923 [eess.IV].

[22] Y. Nagaraj, A. Teja, and D. Narasimha, "Automatic segmentation of intima media complex in carotid ultrasound images using support vector machine," *Arabian Journal for Science and Engineering*, vol. 44, Sep. 2018. DOI: 10.1007/s13369-018-3549-8.

[23] C. Loizou, T. Kasparis, C. Spyrou, and M. Pantzaris, "Integrated system for the complete segmentation of the common carotid artery bifurcation in ultrasound images," vol. 412, Jan. 2013, pp. 292–301. DOI: 10.1007/978-3-642-41142-7_30.

[24] L. Christodoulou, C. P. Loizou, C. Spyrou, T. Kasparis, and M. Pantziaris, "Full-automated system for the segmentation of the common carotid artery in ultrasound images," in *2012 5th International Symposium on Communications, Control and Signal Processing*, 2012, pp. 1–6. DOI: 10.1109/ISCCSP.2012.6217824.

[25] N. Ikeda, N. Dey, A. Sharma, A. Gupta, S. Bose, S. Acharjee, S. Shafique, E. Cuadrado-Godia, T. Araki, L. Saba, J. Laird, A. Nicolaides, and J. Suri, "Automated segmental-imt measurement in thin/thick plaque with bulb presence in carotid ultrasound from multiple scanners: Stroke risk assessment," *Computer methods and programs in biomedicine*, vol. 141, Jan. 2017. DOI: 10.1016/j.cmpb.2017.01.009.

[26] P. Madipalli, S. Kotta, H. Dadi, N. Y., A. C.S., and A. V. Narasimhadhan, "Automatic segmentation of intima media complex in common carotid artery using adaptive wind driven optimization," in *2018 Twenty Fourth National Conference on Communications (NCC)*, 2018, pp. 1–6. DOI: 10.1109/NCC.2018.8600240.

[27] S. S., J. B., R. C., N. Madian, and T. Sunder, "Convolutional neural network for segmentation and measurement of intima media thickness," *Journal of Medical Systems*, vol. 42, Jul. 2018. DOI: 10.1007/s10916-018-1001-y.

[28] J. Y. Shin, N. Tajbakhsh, R. T. Hurst, C. B. Kendall, and J. Liang, "Automating carotid intima-media thickness video interpretation with convolutional neural

networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2526–2535. DOI: 10.1109/CVPR.2016.277.

[29] R. v, J. Joseph, N. P M, and M. Sivaprakasam, "Automated measurement of compression-decompression in arterial diameter and wall thickness by image-free ultrasound," *Computer Methods and Programs in Biomedicine*, vol. 194, p. 105 557, May 2020. DOI: 10.1016/j.cmpb.2020.105557.

[30] M. Biswas, V. Kuppili, T. Araki, D. Edla, E. Godia, L. Saba, H. Suri, T. Omerzu, J. Laird, N. Khanna, A. Nicolaides, and J. Suri, "Deep learning strategy for accurate carotid intima-media thickness measurement: An ultrasound study on japanese diabetic cohort," *Computers in Biology and Medicine*, vol. 98, May 2018. DOI: 10.1016/j.compbiomed.2018.05.014.

[31] S. Savaş, N. Topaloglu, Ö. Kazcı, and P. Koşar, "Classification of carotid artery intima media thickness ultrasound images with deep learning," *Journal of Medical Systems*, vol. 43, Jul. 2019. DOI: 10.1007/s10916-019-1406-2.

[32] C. Qian and X. Yang, "An integrated method for atherosclerotic carotid plaque segmentation in ultrasound image," *Computer Methods and Programs in Biomedicine*, vol. 153, Oct. 2017. DOI: 10.1016/j.cmpb.2017.10.002.

[33] R.-M. Menchón-Lara, J.-L. Sancho-Gómez, and A. Bueno-Crespo, "Early-stage atherosclerosis detection using deep learning over carotid ultrasound images," *Applied Soft Computing*, vol. 49, pp. 616–628, 2016, ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2016.08.055. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494616304574.

[34] B. Khagi and G.-R. Kwon, "Pixel-label-based segmentation of cross-sectional brain mri using simplified segnet architecture-based cnn," *Journal of Healthcare Engineering*, vol. 2018, pp. 1–8, 2018. DOI: 10.1155/2018/3640705.

[35] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017. DOI: 10.1109/tpami.2016.2644615.

APPENDIX A: RESULTS

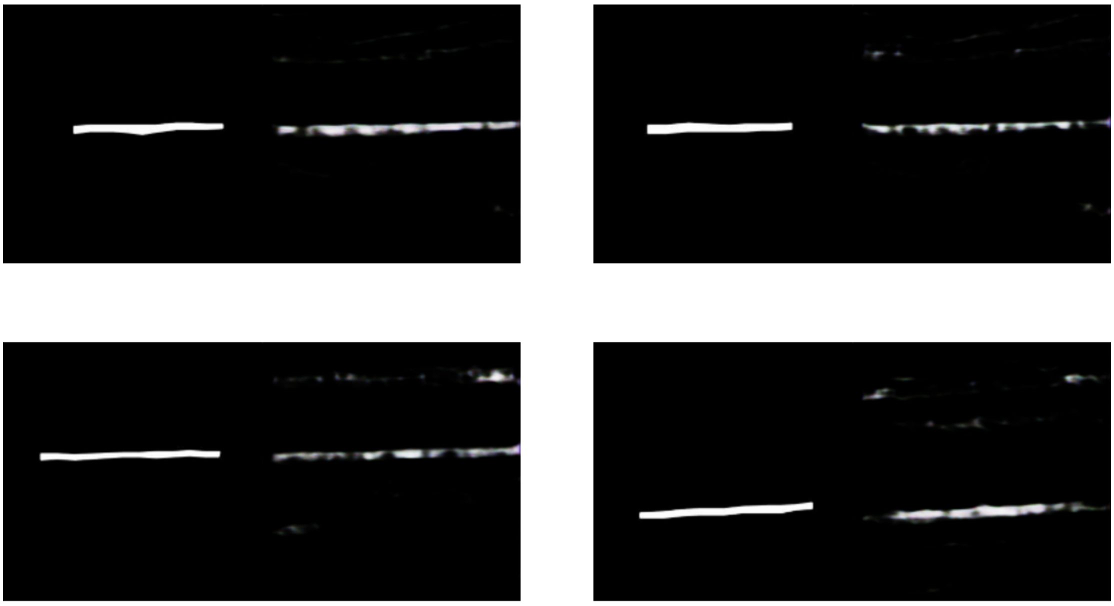

Figure A.1: Sample outputs of the final result and their ground truth

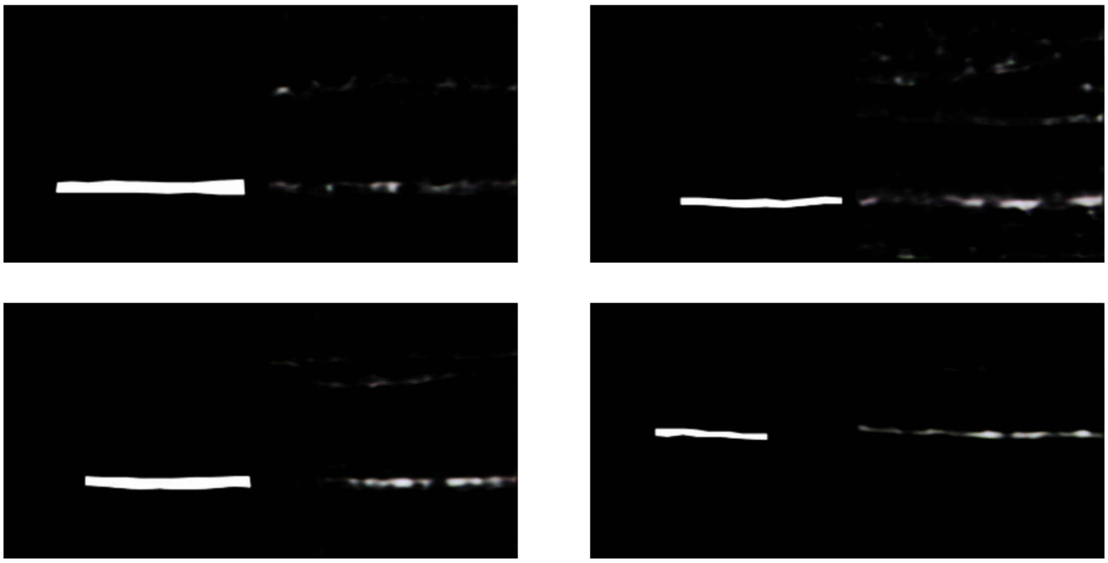

Figure A.2: Sample outputs of trial results using 8 batch size, training 15 epochs, and 5 steps per epoch with their ground truth

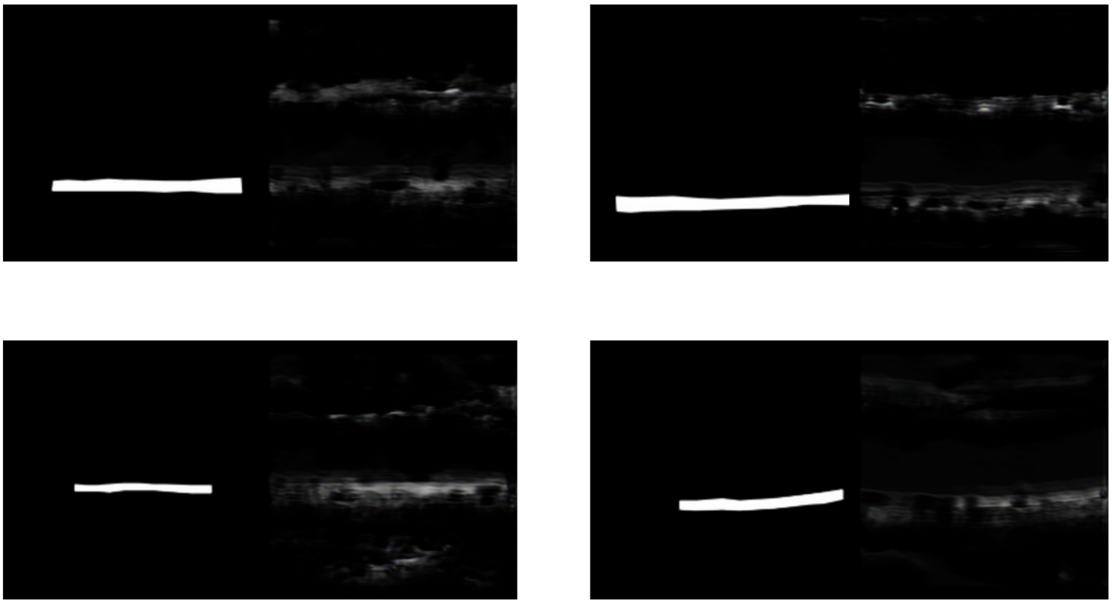

Figure A.3: Sample outputs of trial results using 32 batch size with their ground truth