University of Stavanger

FACULTY OF SCIENCE AND TECHNOLOGY

MASTER'S THESIS

Study programme/specialisation: Master of Science in Computational Engineering	Spring semester, 2021 Open/Confidential: Open
Author: Habib Ullah	
Programme coordinator: Aksel Hiorth Supervisor(s): Ketil Oppedal and Álvaro Fernández Quílez	
Title of master's thesis: Unsupervised Learning for Prostate Tumor Detection	
Credits: 30 ECTS	
Keywords: Variational Autoencoders, Autoencoders, Deep Leaning, T2-weighted, ADC, Magnetic Resonance Imaging, Prostate Lesion Detection.	Number of pages: 69 + supplemental material/other: 22 Stavanger, July 13, 2021

University of Stavanger

Faculty of Science and Technology
Department of Energy Resources (IER)

Unsupervised Learning for Prostrate Tumor Detection

Master's Thesis in Computational Engineering

by

Habib Ullah

Internal Supervisors

Ketil Oppedal

Álvaro Fernández Quílez

July 13, 2021

"The present is theirs; the future for which I really worked, is mine."

Nicola Tesla

Abstract

The detection of a lesion in the prostate can be a challenging task and is crucial for the early diagnosis of Prostate Cancer (PCa). Magnetic Resonance Imaging (MRI) examination provides a comfortable and precise solution to detect prostate lesions. The ability of humans to detect lesions from the prostate MRI by learning from the appearance of healthy prostate structures might help deep learning (DL) architectures achieve the human level's detection ability.

To this end, this thesis proposes an effective method to detect lesions in the patients by learning the distribution of healthy prostate images using auto-encoder-based methods in an unsupervised framework. The thesis methodology involves two main steps: training of DL models, and binary classification of the images as well as detection of lesions in unhealthy images. This work makes use of two DL architectures for the task: Variational Autoencoder (VAE) and Autoencoders (AE), which are then compared in terms of lesion detection and classification ability. The binary classification is based on pixel-wise reconstruction error. The thesis uses the T2w and Apparent Diffusion Coefficient (ADC) MRIs of the prostate, from PROSTATEx Challenge data. The thesis explores the effect of data imbalance in the final results by using two different configurations of test data, balanced and imbalanced data, for both modalities.

The final results indicate that VAE performs significantly better than AE in terms of ROC-AUC, and both models perform notably better for ADC images than T2w images.

Acknowledgements

This thesis marks the end of my Master of Science degree in Computational Engineering at University of Stavanger, Department of Energy Resources (IER). The thesis was conducted during the spring semester of 2021, and has not only been challenging, but also educational and exciting.

I am grateful for the opportunity I have gotten to be able to work with new technology, state-of-the-art hardware at my disposal at the University, and surrounded by people from several disciplines for continuous support. I would like to give a special thanks to my head supervisor Ketil Oppedal and co-supervisor Álvaro Fernández Quílez for their excellent support and guidance during the thesis, and much-appreciated feedback throughout the entire master period. Furthermore, I also want to thank Theodor Ivesdal for help and advice related to the university's UNIX-system.

I would like to thank my program coordinator Professor Aksel Hiorth for his immense support throughout my degree. I would also like to thank all lectures and co-students for two exciting years filled with memories and new knowledge.

Contents

Abbreviations

2D	**T**wo - **D**imensional
3D	**T**hree - **D**imensional
MRI	**M**agnetic **R**esonance **I**maging
T2w	**T**2 **- w**eighted **I**mage
PCa	**P**rostate **C**ancer
PSA	**P**rostate - **S**pecific **A**ntigen
ADC	**A**pparent **D**iffusion **C**oefficient
DWI	**D**iffusion **W**eighted **I**maging
DL	**D**eep **L**earning
NN	**N**eural **N**etworks
CNN	**C**onvolutional **N**eural **N**etworks
VAE	**V**ariational **A**utoencoders
AE	**A**utoencoders
CAE	**C**onvolutional **A**utoencoders
DRE	**D**igital **R**ectum **E**xamination
TRUS	**T**ransrectal **U**ltrasound **S**can
NumPy	**N**umerical **P**ython
LR	**L**earning **R**ate
BS	**B**atch **S**ize
LD	**L**atent **D**imension
PI-RADS	**P**rostate **I**maging **R**eporting and **D**ata **S**ystem
MSE	**M**ean **S**quared **E**rror
MAE	**M**ean **A**bsolute **E**rror
SSIM	**S**tructural **S**imilarity **I**ndex **M**easurement
HCP	**H**uman textbfConnectome **P**roject

KDD	**D**ata **M**ining and **K**Knowledge **D**iscovery
VQ-VAE	**V**ector **Q**uantized - **V**ariational **A**utoencoders
KL	**K**ullback – **L**eibler
ANN	**A**rtificial **N**eural **N**etworks
API	**A**pplication **P**rogramming **I**nterface
OpenCV	**O**pen **S**ource **C**omputer **V**ision
DICOM	**D**igital **I**maging and **C**ommunication in **M**edicine
PI-RADS	**P**rostate **I**maging **R**eporting and **D**ata **S**ystem
ROC	**R**eceiver **O**perating **C**haracteristics
AUC	**A**rea under the **C**urve
UiS	**U**niversity of **S**tavanger
ML	**M**achine **L**earning
ReLU	**R**ectified **L**inear **A**ctivation **F**unction
GPU	**G**raphics **P**rocessing **U**nit
IDE	**I**ntegrated **D**evelopment **E**nviorment

Chapter 1

Introduction

1.1 Motivation

Globally, Prostate Cancer (PCa) is the second most commonly occurring cancer. It is the fifth-leading cause of men's deaths as a result of complications caused by cancer. Approximately 1.2 million people were diagnosed, and 359000 died because of prostate cancer in 2018 [1]. The number is estimated to increase to approx. 2.3 million new cases by 2040, due to population growth and other factors like obesity, among others [2]. The diagnosis of PCa relies on several tests and initial clinical examination by a doctor, which sometimes may be expensive and time-consuming due to the heavy patient load [3]. There is no single definitive test for prostate cancer; however, the general first test is clinical and performed by an assigned General Practitioner (GP). These examinations from GP might be misleading and un-decisive due to a lack of expertise and proper tools in the GP's office. Mostly Prostate-Specific Antigen (PSA) test is used to discriminate between high and low-risk patients. However, PSA testing could lead to unnecessary screening tests and over-treatment [3].

Early detection of prostate lesions can play a critical role in the patient's recovery chances. There are many complications present in the current examination method. Several patients complain of getting an infection after a guided biopsy procedure at the hospital. Magnetic Resonance Imaging (MRI) has proven to be a successful tool to detect and diagnose prostate cancer. Increased use of MRI has made the examination process more comfortable and somehow more efficient, leading to the better examination of the prostate gland and detecting malignant lesions in the prostate glands [4]. The analysis of MRI is a time-consuming task and reader-dependent and leads to variability in the outcome of the task at hand depending on the person in charge of it.

Deep learning (DL) architectures have proven helpful while carrying out the tasks on

various recognition tasks such as image classification and object detection [5]. The training of these models requires a large number of medical data [6]. The human's ability to detect abnormalities in the images after seeing a handful of healthy images raises a thought that what if it is possible to formulate a DL model that can detect lesions in unhealthy images by training only on samples of healthy images. The inherent ability of Variational Autoencoders (VAE) to classify outliers from average data by learning the distribution of healthy data provides a solution to the problem and thus motivates us to achieve this human level of classification and detection ability. The application of deep learning models could undoubtedly improve the existing MRI examination by sparing time for experts while maintaining the quality of diagnosis and thus, allowing GP to consider more patients in the risk group, ultimately referring for further examination and possible treatment.

1.2 Problem Definition

The thesis's primary goal is to use the DL and CNN for prostate cancer MRI lesions detection in an unsupervised framework on the T2-weighted (T2w) and ADC MRIs of the prostate. To the best of my knowledge, this thesis introduces a new approach to classify and detect lesions in prostate MR images, using Variational Autoencoders (VAE) and Autoencoders (AE) in an unsupervised manner.

1.2.1 Objectives

- To explore the use of DL and, in particular, CNN in prostate cancer MRI lesion detection in an unsupervised framework.
- To explore the dataset of T2w & ADC MRIs and concepts of different types of autoencoder structures.
- To compare the results obtained by VAE and AE on two modalities, T2w and ADC.
- To propose and analyze the method of classification for MRI images into two classes, healthy images and unhealthy images.
- To detect the lesions in MRIs of the prostate and analyze the performance of proposed models.

1.2.2 Proposed Method Overview

Firstly, medical imaging data consisting of MRIs of the prostate are loaded. The data is pre-processed and split into healthy and unhealthy data. Then, the model is trained on the unhealthy data, and predictions are made on the test data, which contains both healthy and unhealthy MRIs that are unseen by the model before, using trained models. The classification is based on pixel-wise reconstruction error using mean-squared error (MSE) or Structural Similarity Index Measurement (SSIM) between reconstructed and original images. The basic idea of classification and detection is that the trained model, which have only seen the healthy images, will not be able to reconstruct lesions present in unhealthy images, thus, increasing the pixel-wise reconstruction error between the original and reconstructed images. The results are evaluated and interpreted for better understanding. Figure 1.1 shows an overview of the proposed method.

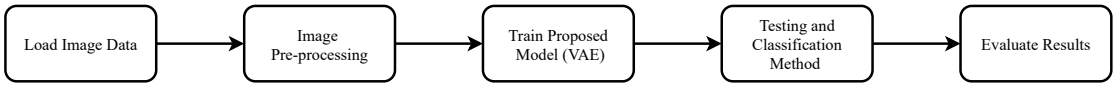


Figure 1.1: A simple overview of the proposed methodology.

1.3 Related Work

The problem of classification and detection of lesions in the tissues is getting considerable attention from the research community. Earlier works such as [7],[8], [9] have suggested successful methods for anomaly detection and prostate image segmentation on brain and prostate MRI images using different methods.

This thesis uses DL models to detect a lesion in T2w and ADC MRI images of the prostate. The primary DL model used in this thesis is Variational Autoencoder (VAE), firstly introduced by Diederik P. Kingma and Max Welling in the paper [10]. The framework has a wide array of applications, from generative modeling, semi-supervised learning to representation learning.

The proposed method used in this thesis is inspired by the work of Xiaoran Chan et al. in the paper [11]. They proposed a simple approach of classifying and detecting lesions in brain MRIs by learning the distribution of healthy images in an unsupervised framework using VAE and AEE with proposed constraints models. The paper addressed the problem of lack of consistency in latent space representation and proposed the addition of a certain constraint during the training process to encourage latent space consistency. The training was performed on the HCP dataset, and the BRATs dataset is used to test the model. Their proposed method was able to give the ROC-AUC of 0.92 during the classification of

two classes, and the model was successfully able to detect the lesion using the proposed method.

Wolter Bulten et al. in their paper [12] proposed an unsupervised method for the classification of prostate tissues by using self-clustering convolutional adversarial autoencoders. The clustering methods are incorporated during the training of the model and being trained on rains on hematoxylin and eosin (HE) input patches. Another scientific paper [13], presented by Rong Yao et al. somewhat proposed a similar concept to be used in this thesis; however, their approach was a comparative one, and comparison of experiments are conducted on KDD CUP 99 dataset and MNIST dataset while using VAE, autoencoders (AE) and Kernel Principle Component Analysis (KPCA) as comparison models. Though the quality of the dataset used in this paper doesn't provide a very good basis for a future study on other complication datasets like PROSTATEx Challenge data comprising prostate MRI images, yet, the consistency of the proposed models may be employed for further experimentation.

Lu Wang et al. in their paper [14] proposed a very sophisticated method of detecting anomalies in the image data by estimating the latent space of autoencoders using a discrete probability model. The improper dimensionality often leads to the reconstruction of unwanted and unhealthy parts. The method first adopts VQ-VAE as a reconstruction model for a discrete latent space of normal samples, followed by using the auto-regressive model Pixel Sail used to outputs the probability model of discrete latent space. The model proposed that the ROC-AUC can be enhanced by over 15% for autoencoders, offering competition to state-of-the-art methods.

1.4 Outline

This thesis begins with a brief introduction and explanation of the motivation behind the concept. The remaining part of the thesis is divided into the following chapters.

- The second chapter, named Medical Background, illustrates the essential medical theory used in the thesis.
- The third chapter, named Technical Background, gives us an insight into the theoretical knowledge of the concepts and techniques used in the thesis.
- The fourth chapter, named Dataset and Image Pre-Processing, describes the dataset and methods for pre-processing of data.
- The fifth chapter is Solution Approach, and it describes the proposed method to detect a lesion in MRI of the prostate.

- Chapter six, named Experimental Evaluation and Results, describes the experimental setup used in the thesis. The chapter also emphasis on the model performance after training on a non-healthy dataset and predictions made by it.
- Chapter seven presents the discussion of achieved results and limitations of the thesis.
- Chapter eight is the last one and presents the conclusions of the work presented in this thesis and recommendations for future work that can be done in this field.

Chapter 2

Medical Background

2.1 Prostate cancer

Globally, Prostate cancer is the second most commonly occurring cancer. It is the fifth-leading cause of men's deaths as a result of complications caused by cancer. Approximately 1.2 million people were diagnosed, and 359000 died because of prostate cancer in 2018 [1]. The number is estimated to increase to approx. 2.3 million new cases by 2040, due to population growth and other factors like obesity, etc.[2].

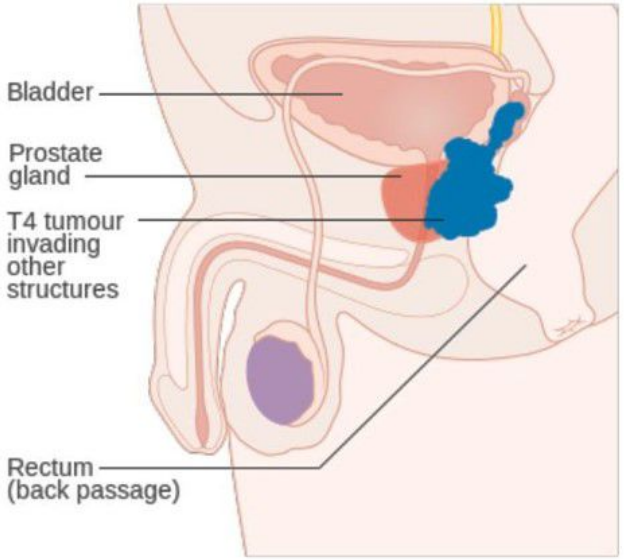


Figure 2.1: Figure shows a stage T4 prostate cancer.
The figure is reprinted in unaltered form from Wikimedia commons, File: Diagram showing stage T4 prostate cancer CRUK 454.svg, licensed under CC BY-SA 4 [15]. 0

Regeneration in tissues in the human body happens because of cell division. New cells take the place of old cells as they die. The growth rates, however, can change in some cases and

cause overproduction, thus leading to cell accumulation, which ends up in the formation of a lesion. These lesions may be benign or malignant. This uncontrolled division of cells often leads to the creation of cancer. Cancer cells can grow into neighboring tissue and expands their roots to different parts of the human body [16]. The prostate is located underneath the bladder in a normal male human. The primary function of the prostate is to produce fluid, which, combining with sperm cells from testicles and fluids from other glands, makes up semen. The prostate gland grows with the age of a human being, often leading to larger prostate problems in men. The exceptional growth of the prostate is not always cancerous; however, it can cause many complications. The most common symptoms of prostate cancer involve frequent urination, forced urination, dripping after urination, and blood in the urine [16].

2.2 Prostate Cancer Examination Methods

Globally, numerous methods are being used to examine the patient and diagnose prostate cancer. The detailed description of typical routines that are being used during the diagnosis process of prostate cancer is explained as follows.

2.2.1 Prostate-Specific Antigen Test

The level of Prostate-Specific Antigen (PSA) is measured in the patient's blood by taking a blood sample of a patient. The amount of PSA usually increases in the blood when a patient has prostate cancer; however, this test has some limitations as PSA can also increase during the benign growth of the lesion. The test can be useful if GP is able to compare the results of tests from the blood sample taken before and after the patient got cancer [16].

2.2.2 Digital Rectum Exam

During Digital Rectum Exam (DRE), GP physically feels the rear part of the prostate by inserting a sanitized and lubricated gloved finger in the patient's rectum. GP determines the size and shape of the prostate based on his medical knowledge. DRE is usually performed by GP before referring the patient to an expert. The demonstration of DRE is explained through figure 2.2 [17].

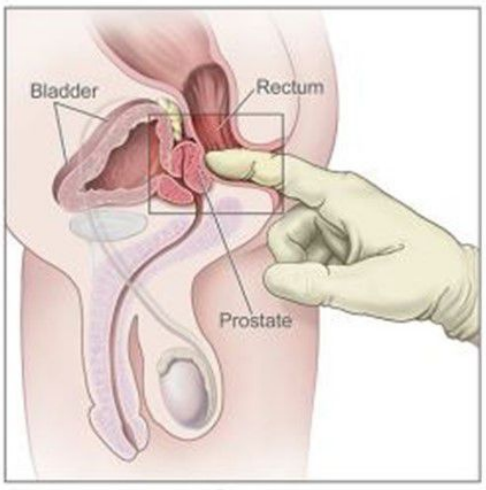


Figure 2.2: Digital rectal examination.
The figure is reprinted in unaltered form from Wikimedia commons, File: 482pxDigital_rectal_exam.jpg [15] [18].

2.2.3 Biopsy

If GP suspects cancer, biopsies are performed as the next examination step. Transrectal ultrasound scan (TRUS) is the most applied method for performing the biopsy. However, biopsy often leads to an infection, and the rate of infection varies between 5% and 7%, despite intensive medical care. A needle is inserted into the prostate eight to ten times to collect tissue samples from several parts of the prostate. The process of TRUS is illustrated in Figure 2.3 . Using the Gleason score grading system, the patient's prognosis is obtained from a biopsy. These scores are evaluated by a pathologist, thus, declaring the stage of prostate cancer. A higher Gleason score indicates higher stage cancer with a poor to a worse prognosis [19].

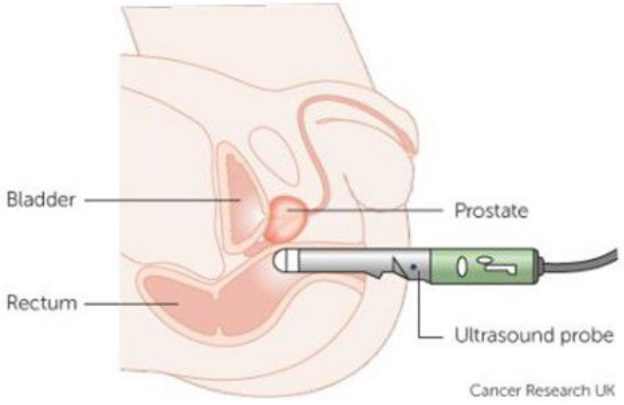


Figure 2.3: Transrectal ultrasound scan examination.
The figure is reprinted in unaltered form from Cancer Research UK's webpage [20] [18].

2.2.4 Magnetic Resonance Imaging

MRI scans are used to find the location of the lesion, and they are usually performed before a biopsy. MRI can also help in determining the nature of the lesion as to whether the lesion is benign or malignant, suggested by some articles [21]. The examination and diagnosis of prostate cancer can be enhanced by a better understanding between medical experts and MRI specialists.

Radiologists interpret MRIs on the basis of Prostate Imaging Reporting and Data Systems (PI-RADS) scoring [22]. PI-RADS was developed to promote the global standardization of prostate multi-parametric magnetic resonance imaging (mp-MRI) examination. This standardization was adopted to improve the detection of clinically significant cancer and distinguish benign lesions to avoid unnecessary biopsies [22].

Chapter 3

Technical Background

This chapter will give general insight into the technical terminologies and concepts used in this thesis while providing the mathematical interpretation behind these concepts.

3.1 Magnetic Resonance Imaging

This chapter starts by elucidating the concepts and terminologies related to Magnetic Resonance (MR) technology and process related to MRI.

3.1.1 Basic Terminologies

In MR-examination, digital images of internal organs are generated by exposing a patient to a strong magnetic field. MRIs are obtained using a pulse sequence that involves adjustable timing values, termed Repetition Time (TR) and Echo Time (TE). TR describes the time between similar events on a recurrent series of pulses and echoes while time separating the center location of RF pulse and the corresponding echo is summarized by TR [23] MRI uses the natural properties of water lipids to capture images. The most fundamental parameters in MRI are characteristics times that are named spin-lattice relaxation time (T1) and spin-spin relaxation time (T2). Longer TR and TE generate T2-weighted images (T2w), while shorter TR and TE spans produce T1-weighted images (T1w). Tissue images with longer T2 specific times are usually brighter and take more time to produce than T1 due to strong signal intensity in T2. The brightest region of T2w images represents fluids, and the grey regions correspond to water- and fat-based tissues. The pathological process can change the natural balance to water content and tissue vascularity. Therefore, the presence of lesion can cause changes in tissues, thus altering

its T1 or T2 natural relaxation rates, and can be interpreted as observable changes in conventional T1w or T2w images. Abnormal brightness on a T2 image indicates a disease process such as trauma, infection, or cancer [23]. Image contrast can also be created using apparent diffusivity. Diffusion-weighted magnetic resonance imaging (DW-MRI) uses measurement diffusion properties of water to construct patterns in MRI images. A combination of images having different amounts of diffusion weighting provides an apparent diffusion coefficient (ADC) map or ADC image. In ADC imaging, protons exhibit free mobility in the lesion region than the surrounding and are often interpreted as signal loss. Therefore, the corresponding area of higher diffusivity is represented as a brighter region, indicating a high ADC value in the obtained ADC map, while infarcted areas appear dark on the ADC map with low ADC values [24]. Figure 3.1 shows the difference between randomly taken T2w and ADC MRI of prostate.

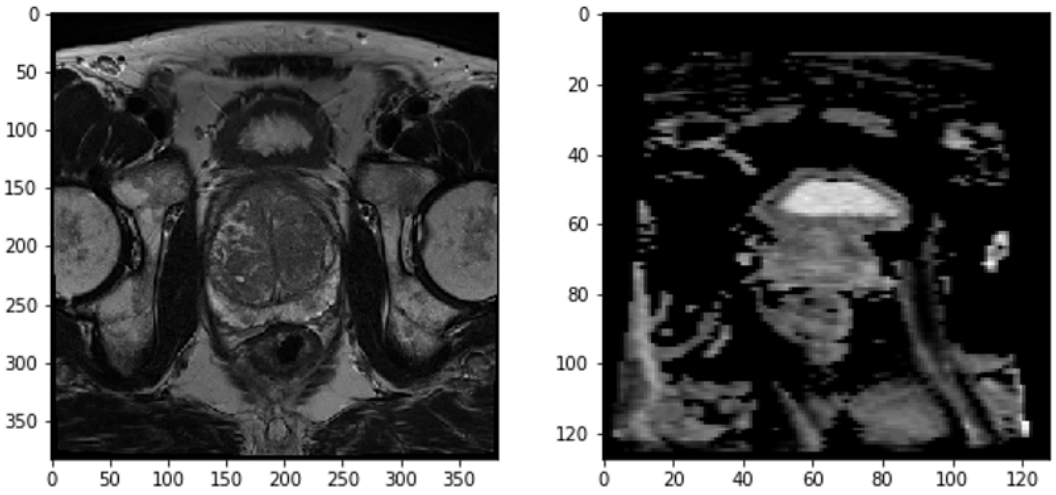


Figure 3.1: Figure shows the random MRI slices of prostate for two modalities, T2w image (left) and ADC image(right)

3.2 Neural Network

The Neural Network (NN) is named after the neural setup of the human brain because NN operates as the system of neurons. A NN is a structure of the different processing elements (often referred to as nodes or neurons) connected with unidirectional signal channels, termed connections. The value is calculated by each neuron, shared via connection to the next layer of neurons. The processing of neurons only depends on the current input values and values that are being stored in the local memory of a neuron [25]

Figure 3.2 shows the concept of a simple feed-forward network with two inputs x = [x1, x2], one hidden layer with four neurons, and two outputs (y). The input value x feeds the

initial information that moves forward to each neuron in the hidden layers, and output (y) is predicted. The mentioned figure 3.2 also shows a neuron with the corresponding mathematical functions. The result from the activation function (g(z)) is equal to the neuron output, where y is equal to the final hidden layer output, as shown in equation 3.1.

$$\hat{y} = g(z) = g\left(\sum_i w_i x_i + b\right) \tag{3.1}$$

Equation 3.1 is the calculation of the last hidden layer, where g is the activation function and z represent neuron input and local parameters.The input variable x in equation 3.1 resembles the previous layer's output value or the network input values if the calculation applies to the first hidden layer.

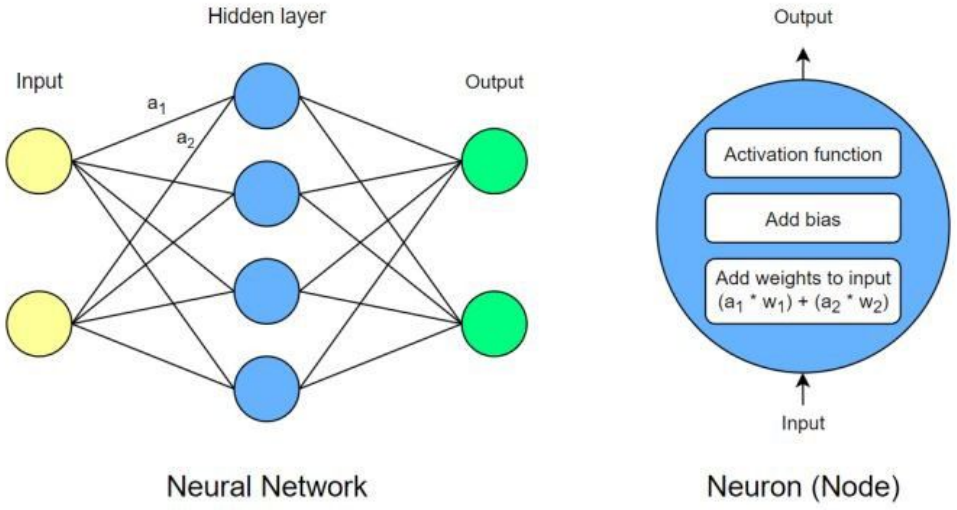


Figure 3.2: Illustration of a neural network on the left side with two input values (x),one hidden layer including 4 neurons, and two output values. Neurons, input, and output values are combined with connections. The right side of the figure illustrates a neuron and the including mathematical functions [25] [18].

3.3 Convolutional Neural Networks (CNN)

Convolution is the mere application of a filter to an input that gives out the activation. Feature map can be obtained by repeated application of the same filter to the document results in the map of activations, suggesting the locations and strength of a detected feature in input, for instance, an image. The convolutions are applied to extract the features that could be missed in simply flattening an image into its pixel values. Convolutional neural networks are like neural networks expect they have at least one convolutional layer in them. In CNN, only the last layer is fully connected, whereas, in

ANN, all the neurons are interconnected with each other. Thus, CNN not only reduces the number of dependent units in the network, but also reduces the chance of over-fitting by learning only fewer parameters. They also offer context information to the small neighbourhood, enabling the network to achieve better predictions in image data. Deep CNN have shown promising value in the task of pattern and visual recognition in recent years [26][27].

"Convolutional networks are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers."
-—Page 326, Deep learning, 2016 [15]

CNN architecture may contain one or more than one model that takes an input image and outputs the prediction by passing information through one or more convolutional or other image processing layers. This section will briefly explain some of the terms related to CNN.

3.3.1 Convolutional Layer

A standard convolutional layer accepts an input image of size n*m, where n is the number of pixels in width and m is the height. The operation of the convolutional layer depends upon two important parameters, the number of paddings (p) and strides (s). The padding adds a boundary around the image where the default value of padding is zero. Padding is usually added when kernel size and input size don't add up. The number of n or m to shift the kernel to move across the image is decided by the number of strides. A kernel extract features, like edges and corners, from the receptive field and outputs a feature map. The spatial dimensions of the output image from the convolutional layer are usually less than the input but can be equal if the stride is equal to 1 [15].

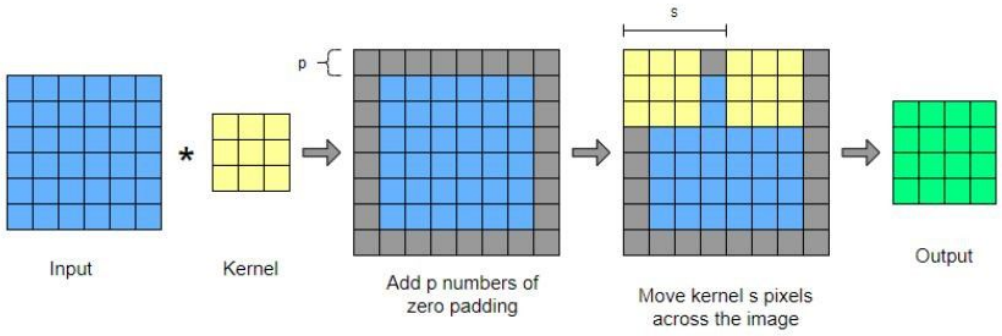


Figure 3.3: Illustration of the process behind a convolution layer [15] [18].

Figure 3.3 illustrates the standard process of the convolutional layer, where the input is an image of n and m equal to 6 pixels.

3.3.2 Transposed Convolutional Layer

The transposed convolutional layer is used during the up sampling of an image. Figure 3.4 demonstrates the process of transposed convolution layer. The output of the transposed convolutional layer is also controlled by strides and padding. The output from transposed convolutional layer has greater spatial dimensions, unlike the convolutional layer. The transposed convolution is not opposed to the convolution in terms of values, but it only reverses the spatial dimensions of standard convolution [28].

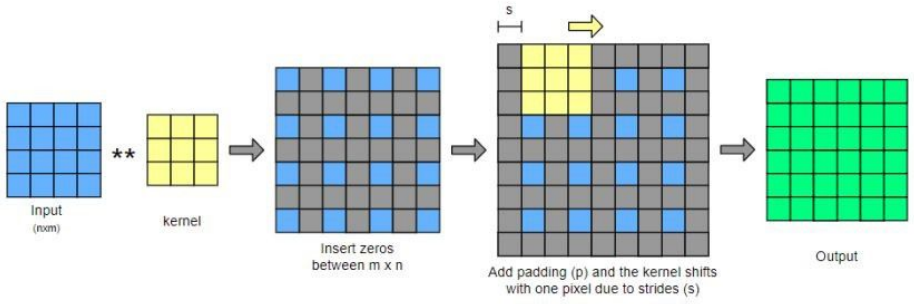


Figure 3.4: Illustration of the process behind a transposed convolution layer [28] [18].

3.3.3 Dense Layer

A Dense Layer consists of neurons from a NN and is mostly added for classification problems. A dense layer is part of the Keras library [29] and consists of two or more nodes. A decision boundary classifies samples from a vector space into two classes. This layer is quite beneficial for the fact that it can only draw one decision boundary. The number of neurons (N) in a dense layer depends upon the number of classes in the output [30].

Figure 3.5 illustrates three examples where two are possible to separate with one decision boundary, and one is not. The illustration visualizes an AND gate, OR gate, and an XOR gate. Since one node draws one decision boundary, two nodes are required to draw the decision boundaries for an XOR gate. The number of neurons (N) in a dense layer is equal to the number of classes in the output. For classification on values reaching from 0 to 9, N is equal to ten [18].

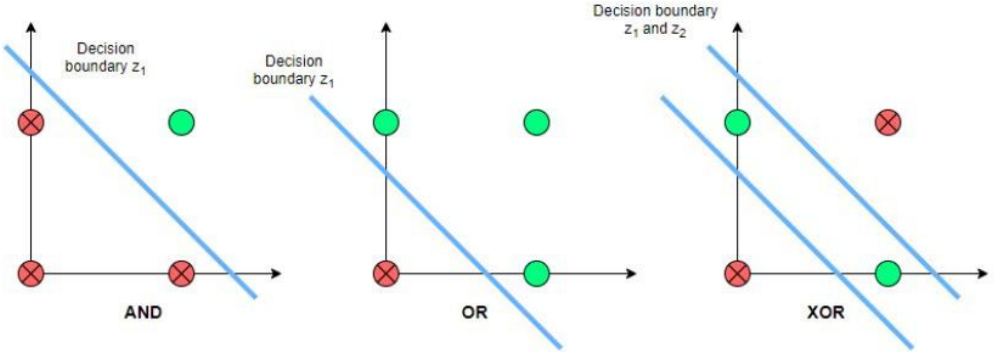


Figure 3.5: Shows classification problems where two is possible to separate with one decision boundary and one needs a dense layer as it is impossible to classify using one decision boundary [15] [18].

3.4 Augmentation

Most of the DL methods are data-dependent and require a huge amount of data to make them work efficiently. The availability of enough medical data is a huge problem due to certain factors like patient privacy or sensitive information when it comes to the usage of deep learning models in the medical field. Therefore, to solve this problem, a technique called augmentation is mostly used to extend the dataset artificially by performing certain actions on the dataset. The image augmentation doesn't make new images in the dataset but provides the new version of the same dataset. Different implementations are accessible for use; some of the important image augmentation types are briefly explained as follows [31].

- Image shift: Moves all the image pixels in one direction; mostly horizontal or vertical shits are considered. The image dimensions remain the same after and before shifting.
- Image Rotation: Rotates the image clockwise in the range of 0 and 1. In rotation, pixels will most likely be rotated out of the image frame, leaving blank spaces in a frame that must be filled in.
- Horizontal or Vertical Flip: Reverses all rows or columns pixels, respectively.
- Blur: Randomly blurring the whole image or a certain part of the image.

3.5 Autoencoder

Autoencoders (AE) are basic learning circuits that try to convert inputs into outputs with as little distortion as possible. They serve a vital role in machine learning, despite their simple architecture. Autoencoders were first introduced by Hinton and the PDP group in the 1980s [32]. Their aim was to solve the back-propagation problem without a teacher by only relying on the input data as a teacher. Autoencoders provides one of the fundamental principles for unsupervised learning by working together with Hebbian learning rules [32]. In autoencoders, we efficiently compress the provided input and encode the data so that it can be decoded back from the compressed representation or code of data, often called as the bottleneck, to get the representation that is close to the original input. Autoencoders have the in-built ability to compress the data efficiently by extracting the important features and ignoring the noise in the data. Autoencoders can only output the data that they have been trained on, making them data-specific and the outputs of the autoencoders are mostly degraded in comparison to original inputs.

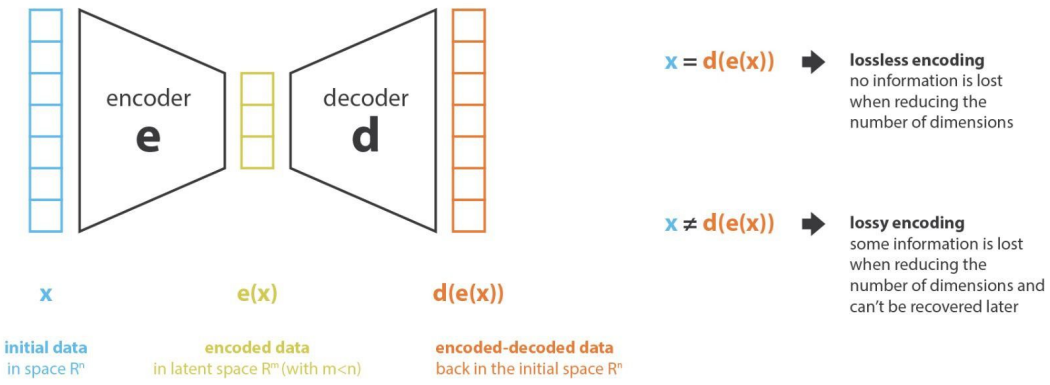


Figure 3.6: Illustration of basic principle of Autoencoder with encoder and decoder [33].

Figure 3.6 shows the typical demonstration of Autoencoder, where the encoder is the feedforward NN which compress the data into latent space representation, code or bottleneck represent the reduced representation of input data and decoder is also a feedforward NN like the encoder; however, it up samples and reconstructs the input back to original dimensions from the compressed data representation in code.

3.5.1 Convolutional Autoencoders

Convolutional Autoencoders (CAE) works as the simple AE; however, the encoding and decoding layers are called convolution and deconvolution layers, respectively. The deconvolution layers are sometimes also interpreted as up sampling or transpose convolution. CAE learns by encoding the input into a group of simple signals and reconstructing the input from those signals. Moreover, image geometry or generation of reflectance of the image can also be modified in CAE [34].

3.6 Variational Autoencoders

Variational Autoencoders (VAEs) were introduced by Diederik P. Kingma and Max Welling in the paper [10]. The framework of variational autoencoders (VAEs) in paper [10] provides a principled method for learning deep latent-variable models and corresponding inference models using stochastic gradient descent. VAEs can produce new images like AE; however, instead of producing a single value for each latent variable independently, the encoders in VAEs generate the probability distribution for each latent variable [33]. The probability distribution of the latent variable in VAEs are much closer to the training data as compared to the simple AE. In VAEs, the mean and variance of variables in latent space are calculated for each sample, and standard normal distribution is followed. Thus, the points that are closed to each other are representing a similar data sample (same classes). Due to the variational approach in latent representation learning, an additional loss component, Kullbeck - Leibler (KL) divergence, and a specific estimator called the Stochastic Gradient Variational Bayes (SGVB) is incorporated in VAEs.

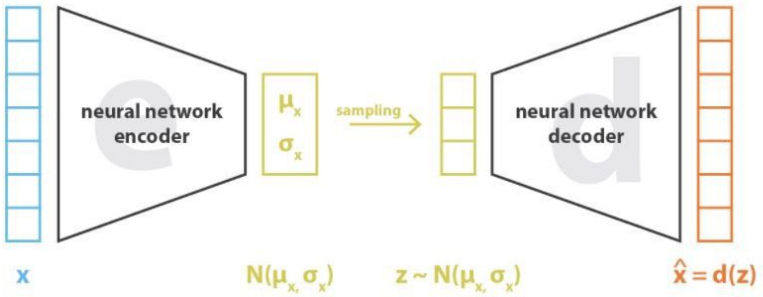


Figure 3.7: Illustration of basic principle of Variational Autoencoder with encoder, latent space representation and decoder [33].

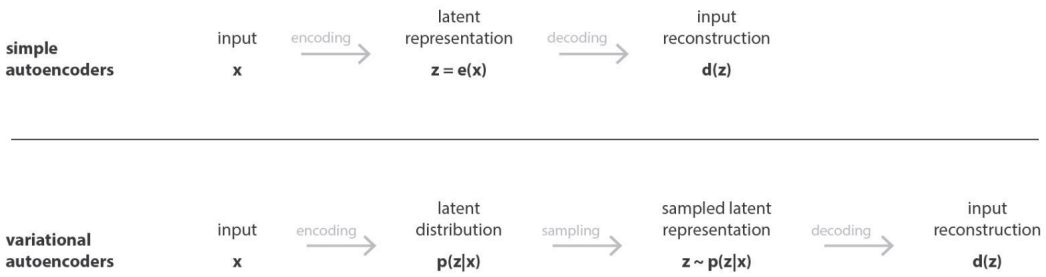


Figure 3.8: Shows the difference between Autoencoder (deterministic) and Variational Autoencoder (probabilistic) [33].

Figure 3.7 and 3.8 shows the simple implementation of VAE and the difference between AE and VAE approach respectively. In VAE, the encoder is the feedforward NN which compress the data into latent space representation, code or bottleneck represent the reduced representation of input data and decoder is also a feedforward NN like the encoder; however, unlike AE, VAE learns from the probability distribution of latent vectors in latent dimensions by using mean and standard deviation of data distributions. The decoder reconstructs the input back to original dimensions from the compressed data representation in code.

3.7 Loss Function

Loss function, often termed as error or cost function, is used to reduce a loss value for the optimization of the model. Loss functions play a vital role in any statistical model. The loss function is responsible for defining an objective against which the model is evaluated, and the parameters learned by the model are selected by minimizing the loss function.

The AE only relies on the difference of reconstructions and original images and thus, only considers individual loss like MSE or MAE loss for the loss function. On the other hand, the variational interference in VAE forces it to include another loss component, KL divergence loss to consider the probability distribution of variables in latent representation. The reconstruction term corresponds to squared error, like in an ordinary AE. The KL term regularizes the representation by encouraging z to be more stochastic. Both DL models AE and VAE are evaluated by taking into consideration the reconstruction loss during the training of DL models.

3.7.1 Kullback–Leibler divergence loss

Kullback–Leibler (KL) divergence loss provides the measure of the difference between two probability distributions. The KL-divergence loss is mathematically represented by equation -. The inclusion of KL divergence in the loss function ensures the learned distribution is very close to the original distribution of latent vectors, which is already considered as a normal distribution [35]. The final objective loss is, therefore, considered as a combination of reconstruction loss and KL-divergence loss.

$$[Q(z|X)||P(z|X)] = E[logQ(z|X) - logP(z|X)] \tag{3.2}$$

In equation 3.2, Q(z|X) is the estimated distribution of data, P(z|X) is the actual distribution of data into latent space, z represents latent variable, and X is the data to be used.

3.7.2 Mean Squared Error

The Mean Squared Error (MSE or L2) calculates the squared difference between the actual and expected values. It is based on the square of Euclidean distance; therefore, it is always positive as error tends to approach zero eventually. The lower the value of MSE, the higher the accuracy of a regression model [36]. Mathematically, MSE is calculated in equation 3.3, where Y_i are the observed values of a variable and $\hat{Y}_i$ are the predicted values of a variable.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{3.3}$$

3.7.3 Structural Similarity Index Measure

Structural similarity index measure (SSIM) is a loss used to measure the similarity between two images. It is usually interpreted as a quality measure of reconstructed image compared to the original image; however, the original image is considered to have perfect quality. In comparison to L2 loss, the SSIM index is a better image quality measure as it is better suited to the human visual system. The SSIM varies with the different distortion in an image; however, L2 or MSE always remains the same for the image, thus making SSIM a superior candidate over other L2 [37].

3.8 Software

The programming language used in this thesis is Python. Python is high level, multi-purpose programming language. The object-oriented programming approach in Python makes it easier to code clearly and logically within the range of small to large-scale projects with easy code readability [38].

Python has a comprehensive library setup that ranges from scientific applications to web-development services. Python also uses many external libraries in addition to built-in libraries. This chapter will explain some of the main libraries used in this thesis.

3.8.1 Tensorflow

Tensorflow is an open-source library used for the implementation of machine learning. This library has a wide range of applications; however, mostly used for the implementation of deep NN algorithms, like training algorithms [39].

3.8.2 Keras

This thesis uses Keras, which is a DL learning application programming interface (API) for humans, providing a Python interface for ANN. Keras provides an interface for the Tensorflow to enhance the fast processing of DL models [29].

3.8.3 Numerical Python

The Numerical Python (NumPy) library is introduced into the Python programming language to analyze and implement high-level scientific computing and data analysis of numerical data and multi-dimensional arrays. This library is being used for many tasks ranging from generating random integers or arrays to advanced mathematical functions. NumPy is also employed to use by other libraries like Tensorflow to generate Tensor objects and more [40].

3.8.4 OpenCV

OpenCV is an open-source library primarily used for computer vision, image processing, and machine learning. The usage of this library in this thesis allows the performing of several actions on the medical images, ranging from image pre-processing to image

augmentation. When this library is coupled with other various libraries, such as NumPy, Python can process OpenCV array structures [41].

Chapter 4

Dataset and Image Pre-Processing

4.1 Dataset

The dataset of prostate MRI used in this thesis is a part of PROSTATEx Challenge data and was collected by performing a clinical examination, and MRI scans at the Radboud University Medical Centre (Radboudumc), Netherlands, in the Prostate MR Reference Center under the supervision of prof. Dr Barentsz. The dataset was collected and curated for research in computer-aided prostate MR diagnosis under Dr Huisman, Radboudumc [42].The two different Siemens 3T MR scanners, the MAGNETOM Trio and Skyra, were used to collect the images. No endorectal coil was used in the acquiring of the images. Figure 4.1 shows a randomly taken T2w MRI slice with a corresponding mask from the PROSTATEx dataset.

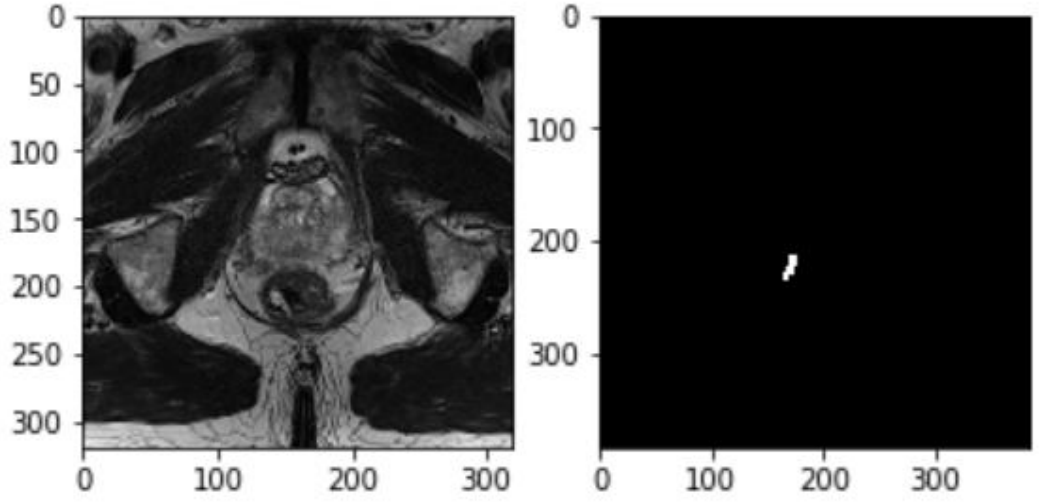


Figure 4.1: A random T2-weighted MRI and the corresponding segmentation mask from the PROSTATEx Challenge dataset.

The PROSTATEx challenge aimed to focus on the quantitative methods for the medical images analysis to classify the clinically significant prostate cancer, and it was held in conjunction with the 2017 SPIE Medical Imaging Symposium. The relevant dataset used in this thesis contains 201 subjects, split into training, testing, and validation data. The details of the subjects and the corresponding slices are mentioned in table 4.1, and the details of the stratification of data are explained in section 4.2.4 of this chapter. Every case has a T2w and ADC MRI of one anonymous patient's prostate and a corresponding label. The images have all kinds of stored information in the metadata, like name, age, slice thickness, etc. The mask of each MRI case provides information on the location, size, and shape of the prostate lesion present in that case.

MRIs relate to a bundle of 2D images that adds up to show three-dimensional (3D) images. (see chapter 3.1) Due to varying data protocols, changing parameters in data makes it acceptable in medical clinics worldwide. Different datasets are available in The Cancer Imaging Archive (TCIA), containing various medical prostate MRIs. These MRIs are stored in DICOM files and many other formats like (MHD/RAW). However, the data in this thesis is using the DICOM format of medical images [43].

4.2 Image Pre-Processing

This section provides a discussion about the pre-processing techniques used in this thesis. The pre-processing of a dataset is inspired by the work presented on image data in Data Science Bowl held in 2017 by Booz Allen Hamilton and Kaggle [44]. The operations performed for pre-processing of MR images data are explained as follows.

4.2.1 Data Loading

Initially, the required images are downloaded from TCIA. The dataset is opened using NBIA Data Retriever to retrieve the required DICOM images. Moreover, these images are then loaded and analyzed for further processing in Jupyter Notebook by using the Pydicom library to work with Dicom files in Python. Figure 4.2 shows the random slices for two modalities, T2w and ADC.

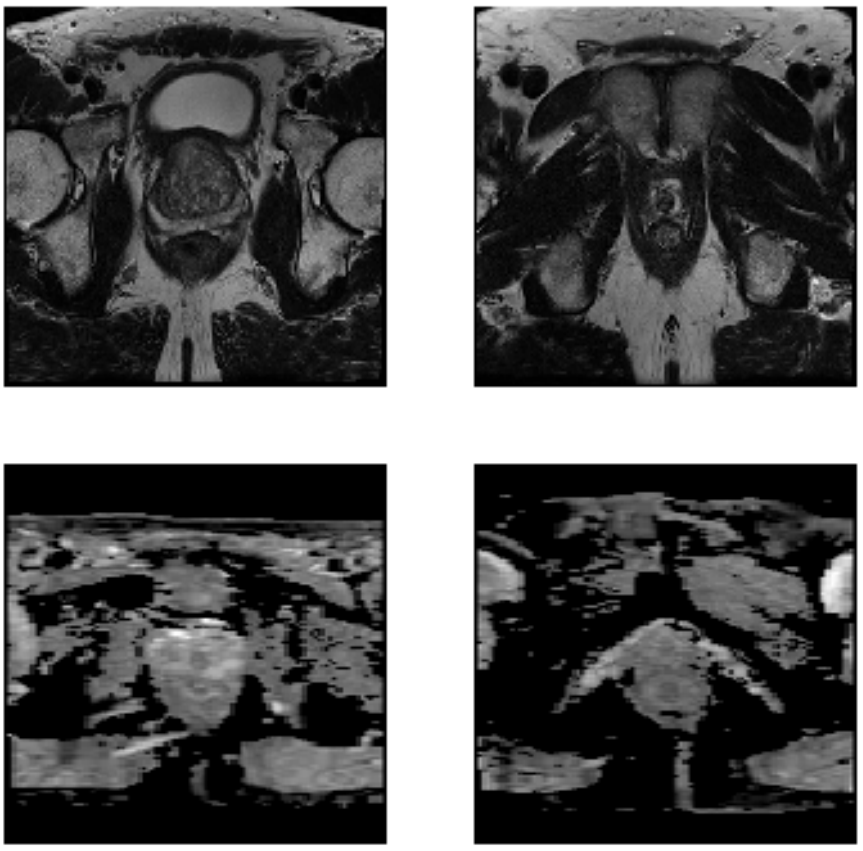


Figure 4.2: Shows the random MRI slices of prostate for two modalities, T2w (first row) and ADC (second row)

Table 4.1 shows the details of the loaded images with the number of slices with respect to the dimensions of slices for both modalities.

Data Specifications	T2w	T2w	T2w	T2w	ADC	ADC	ADC
Width (pixels)	280	320	384	620	75	84	128
Height (pixels)	280	320	384	620	128	128	128
Number of slices	19	220	5708	13	162	5640	100

Table 4.1: Table showing the number of MRI slices and the correlated image sizes for two modalities, T2w and ADC.

The masks for each case are present in the nii or NIfTI format, primarily used for imaging informatics for neuroimaging. These masks are extracted and loaded using the nibabel

library in Python. The information of the masks for each case is present in the csv file, based on whether the mask is clinically significant or not.

4.2.2 Data Filtering

The image-list.csv file contains the information about the clinically significant images or slices of all the images or slices present for every case. The relevant significant information for each case is present in this CSV file for T2w and ADC images. The regex library filters the dataset to extract the relevant images for each case based on the extracted information from the CSV file. The process is performed for both T2w and ADC images datasets. The filtered data is then stored and copied separately with the original names given in the dataset; that is, every case is named as ProstateX-[num] where num is in the range of [0000 , 0204] and the vital information for case numbers 52, 82, and 138 are missing in the dataset.

4.2.3 Data Reshaping

A CNN must train on the images with similar dimensions. Images of different dimensions can be used to train the same network, but not simultaneously. The CNN should be designed to fit the dimensions of images in a dataset to get better results. The dataset comprising of T2w images has height, and weight ranging from 280×280 to 620×620 and is reshaped to 384×384 using OpenCV built-in resize function. The same procedure was followed for ADC images data; however, their dimensions range from 75×75 to 128×128. All the ADC images are reshaped to the image size of 128×128. The channel corresponds to the depth of the image and is set to one for all the slices in the dataset as all the images in the dataset are grayscale. Table 4.2 gives the details of total slices for each modality and their respective dimensions after reshaping.

Reshaped Data Specifications	**ADC**	**T2w**
Total cases	201	201
Total number of slices	5822	5962
Final reshaped width (Pixels)	128	384
Final reshaped height (Pixels)	128	384

Table 4.2: Table showing number of slices present in two modalities, T2w and ADC and their respective reshaped sizes.

4.2.4 Image Stratification and Data Organization

In this thesis, the total subjects used are 201, and 5962 slices are present in the T2w image dataset for all the subjects. This dataset contains both the unhealthy and healthy slices of the subjects. Unhealthy subjects are those slices that contain lesions in them. The slices are sorted and stored in two different arrays based on the presence of lesions in them. Then, the array containing healthy slices is divided into three different arrays for the formation of training, testing and validation datasets. Around 70% healthy are present in the training dataset, 20% in testing, and only 10% of these slices are stored for validation purposes. The unhealthy slices for all subjects are also split into two arrays with a percentage ratio of 80:20, primarily to add them to testing and validation datasets. These arrays made from unhealthy slices are then concatenated to the testing and validation datasets, initially obtained from the division of healthy slices. Finally, three different arrays are obtained; the first one contains a training dataset comprised of only healthy slices of all the subjects, the other two are testing and validation dataset which contains both unhealthy and healthy slices of the subjects. During the stratification, it is strictly considered to put all the slices of one patient in one dataset to avoid leaking lesions between different datasets. Therefore, the training is only performed on the healthy slices of subjects, and predictions are made on the dataset containing unseen healthy and unhealthy slices.

The same procedure is followed for the ADC images, and datasets are obtained similarly to T2w images; the only difference is the total number of slices in the ADC dataset, that is, 5822. Table 4.3 shows the details of slice stratification and shapes of those slices for both modalities T2w and ADC.

Details of Dataset	**T2w**		**ADC**	
	Total slices	**Shape of slices**	**Total slices**	**Shape of slices**
Train Dataset	3637	384×384	3630	128×128
Test Dataset	2066	384×384	1935	128×128
Validation Dataset	259	384×384	257	128×128

Table 4.3: Table showing the details of image stratification and the respective shapes of slices for both modalities, T2w and ADC.

4.2.5 Data Normalizaton

Neural Networks (NN) usually calculate small weights to proceed with input images. The pixel values in most of the images are integers, ranging from 0 to 255. The larger pixel values can make the learning process slow and can cost computing efficiency. Therefore, it

is often considered to normalize the pixel values in the range of [0,1]. The normalization can also help standardize the distribution of pixels to Normal or Gaussian distribution if normalization is done by the standard deviation [45]. In this thesis, normalization is achieved using the equation.

$$Normalization = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{4.1}$$

In equation 4.1, X is the original image pixels, Xmin is the minimum pixel value, and Xmax is the maximum pixel value of the image. The images are normalized to have a range of [0,1]. The data is also standardized to have unit variance, and zero mean.

4.2.6 Saving Organized Data

In this thesis, the NumPy library saves the organized data as a four-dimensional NumPy array in a file with an extension. npy. The first index of the array represents the number of slices present for all the subjects. The following two indexes represent the height and width of the images. The last and fourth index represents the depth of the image, that is the number of channels. In this thesis, the number of the channel is equal to one as all the images present are grayscale. Therefore, the input shape of the stored NumPy array in the DL architecture is (number of slices, height, width, channels).

Chapter 5

Solution Approach

5.1 Introduction

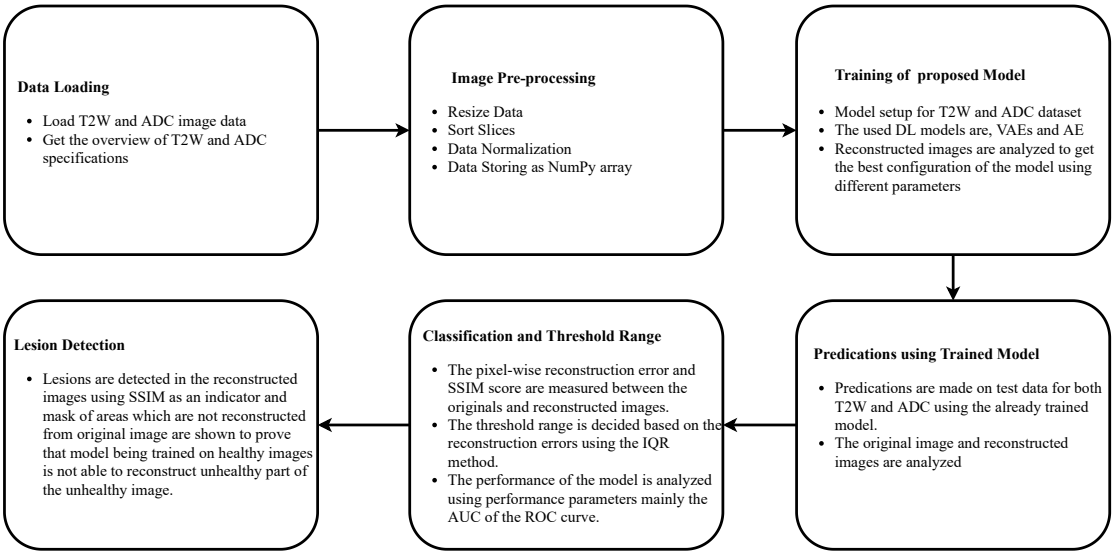


Figure 5.1: A detailed overview of the proposed methodology

5.2 Proposed Method

The existing approaches [11], [12], [13] and [14] for unsupervised classification and detection of the lesions or anomalies in medical images using deep generative models are summarized in section 1.3 of this thesis. This thesis proposes a similar method for unsupervised anomaly detection like the one proposed in the paper [11]. However, in this thesis, the primary DL model used for anomaly detection is VAE compared to AEE as the primary model in the paper [11], and no constraints were applied to the latent space representation in this thesis. The thesis also uses another DL model, AE, for the

comparative study of results. Moreover, the thesis uses the testing dataset that comprises both the healthy images and the unhealthy images of the subjects, all of which are unseen by the model, in contrast to the paper [11], where only unhealthy or abnormal images are used for testing.

The thesis uses a two-stage methodology to carry out the unsupervised lesion detection task. The main idea of the thesis is to classify the images into healthy and unhealthy images by learning the probability distribution of healthy images in the latent space. In the first stage, the data is pre-processed and input into DL models, VAE and AE, for training. The training is performed on the training dataset containing only healthy images of the subjects and cross-validated against the validation dataset. The trained model is then used to make predictions on test data, which contains the combination of the healthy and unhealthy images (see section 4.2.4), to get the reconstructions of test images. In the next stage, pixel-wise reconstruction error is measured by calculating the MSE loss or SSIM score between the original and reconstructed images. Based on these reconstruction errors, the optimal threshold is selected to classify the unhealthy slices from the healthy slices in terms of the reconstruction error. ROC-AUC and PR curves, the overall classification accuracy of the model, and other parameters are calculated to measure the performance of the proposed model. Lastly, the lesions are detected by comparing the SSIM score of the unhealthy or lesion region and the whole image, as the lesion region will output a lower SSIM score than the image. The proposed method is applied to both T2w and ADC datasets for a comparative study.

5.2.1 Experimental Setup

VAE is the primary DL method used in this thesis to classify and detect a lesion in the prostate. VAE, being able to generate high-quality images and having the probability of distribution of latent variables much closer to the original data, is selected as the primary DL model in this thesis. VAE is trained for 1500 epochs, where each epoch uses 31 seconds on average, and the training process is performed on the University of Stavanger GPU gorina6 servers using Nvidia Tesla V1000 with 32 GB memory. Though the network is trained for 1500 epochs, it shares the visuals examples and model weights when the MSE metric for the model improves. The batch size of 32, 64, and 128 samples are used. These batch sizes are used due to the high resolution of T2w images, which makes the training process noticeably slow, and the fact that model performance is not affected by increased batch sizes beyond 128 samples. All models use Adam optimizer with two different learning rates of 0.001 and 0.0001.

Moreover, another DL model, AE, is also used in this thesis to compare results with VAE and evaluate the performance of the primary VAE model used in this thesis. The AE were trained for 200 epochs with batch sizes of 64 and 128 samples. The MSE or L2 is used as a reconstruction loss in AE. The model is using Adam as an optimizer with two different learning rates of 0.001 and 0.0001. This model is also trained on the University of Stavanger gorina6 GPU servers using Nvidia Tesla V100 with 32 GB memory.

The primary language used in this thesis is Python programming language. The DL models are built using Tensorflow [39] and Keras [29]. The classification program is written in Python, and Jupyter Notebook is used as IDE for the programming. The GPU servers are Linux-based servers, and a virtual machine named No Machine is used to access the servers from home. All the models with different configurations are trained, and the whole model with weight is stored in a single file with an extension .h5 using the save command from NumPy library; these trained models are loaded using NumPy load and used for the classification. The classification is done on the personal laptop HP Envy x360 with specifications of Ryzen 7 3700U and 16GB of memory.

5.3 AE Model Design

In this thesis, a simple and basic implementation of the AE model is used. Figures 5.2 and 5.3 show the overview of proposed encoder and decoder structures for AE with the information of all involved layers.

The first layer of the encoder is input with the images of the shape, similar to the VAE model. The encoder of AE consists of four dense layers and four activation layers. ReLU is used as an activation function in all activation layers. The latent dimension or space has a size of 125 units. The decoder network consists of four dense layers and three activation layers, with the last dense layer is the decoder network's output layer. The MSE or L2 is considered as the loss function in this model. The model is simply compiled with L2 loss as reconstruction loss, and Adam is used as an optimizer for the proposed model.The detailed structure of AE with all involved layers and their respective shapes for both modalities are documented in Appendix A.

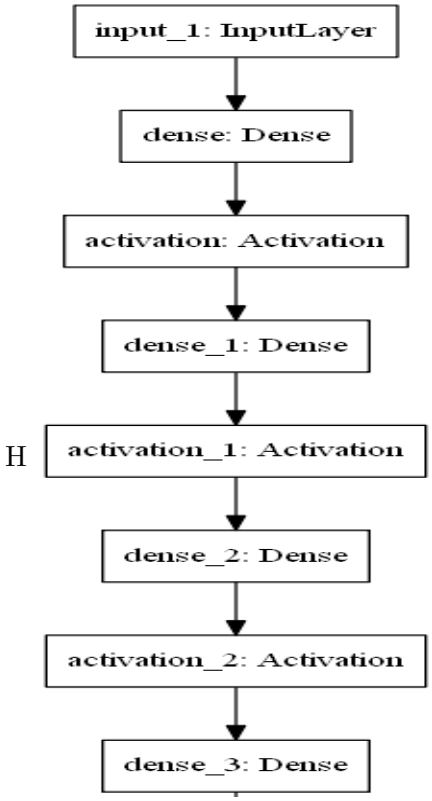


Figure 5.2: Proposed Encoder Network for AE

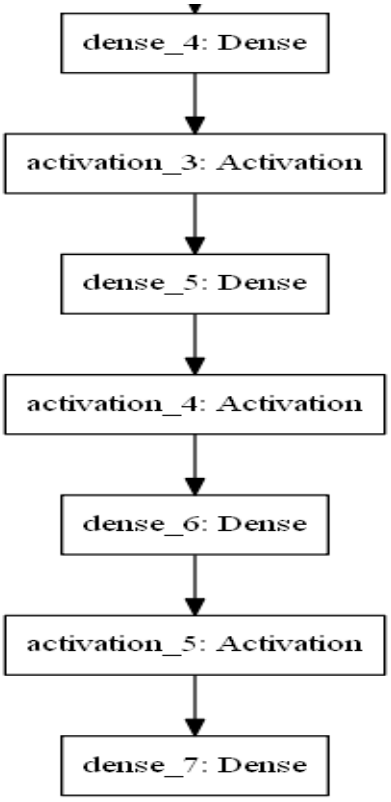


Figure 5.3: Proposed Decoder Network for AE

5.4 VAE Model Design

The VAE implementation is inspired by the work of Rong Yao et al. by using VAE on the MNIST dataset [13] and the original VAE model [10]. Rong Yao et al. work on the VAE generates images of 28×28 pixels, while the paper [11] generates the images of pixel size of 32×32. This thesis is based on these previous implementations, but the architecture is improved and modified to generate images of 384×384 pixels for T2w images and 128*128 pixels for ADC images. The VAE model used in the thesis consists of three parts: the encoder, the latent dimension or space, and the decoder (see section 3.6). Figures 5.4 and 5.5 show the overview of proposed encoder and decoder structures for VAE with the information of all involved layers for both networks, whereas the detailed structures of proposed encoder and decoder for VAE with the respective shapes of each layer for both modalities are documented in Appendix A.

The encoder is input with the data having a shape of 384×384 for T2w and 128×128 for ADC images. Overall, three convolutional layers, one flatten layer, and one dense layer is used in the encoder's architecture. The convolutional layers use the kernel size of (3,3) and strides of size (2,2). The strided convolutional layers are used to allow

learning of weights while down-sampling instead of pooling processes. The input images are down-sampled to the pixel size of 96×96 for T2w and 32×32 for ADC. All the convolutional layers are using ReLU activation. The distribution mean and variance are calculated to randomly sample variables from the distribution in random batch sizes. These sampled batches are then used to extract features from the images, and depending upon the size of latent dimension; these features are stored into the latent dimensions, which is the latent representation of the original images. The dimensions of latent space depend upon the input of the model and desired compressibility in the task. In the case of T2w images, the size of a latent dimension is considered as 128 units and 64 units for ADC images (see section 6.2.6).

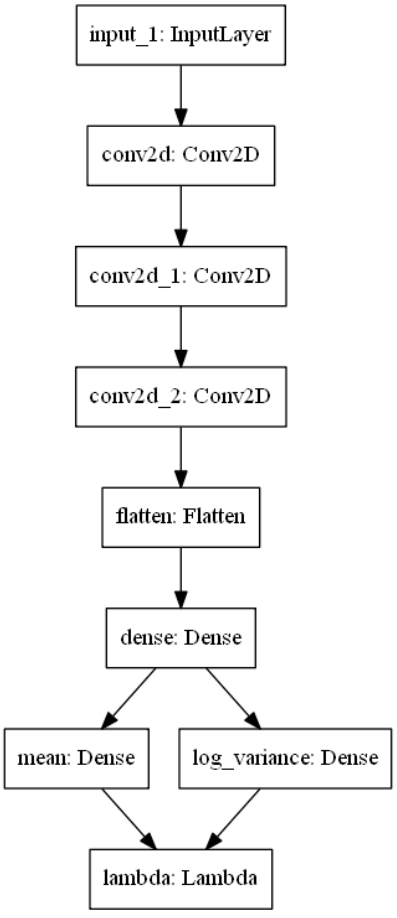


Figure 5.4: Proposed Encoder Network for VAE

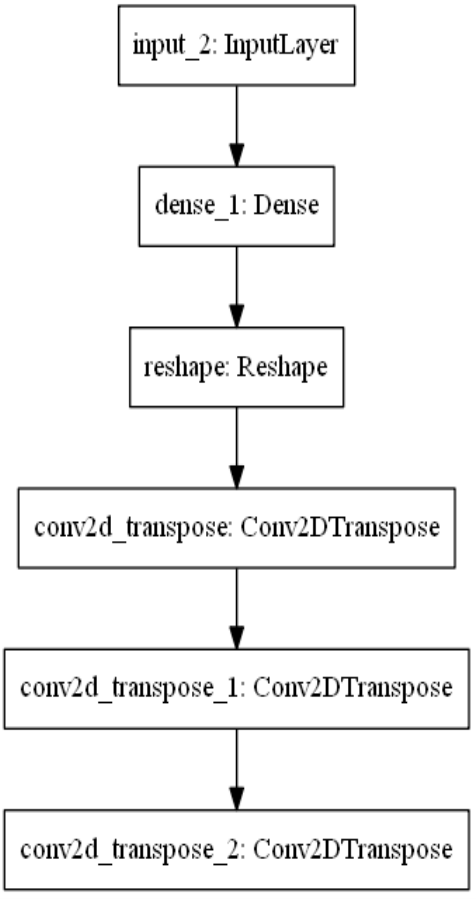


Figure 5.5: Proposed Decoder Network for VAE

The decoder network is used to up-sample and reconstructs the images from the latent space of VAE (see section 3.6). The decoder network used in this thesis consists of one dense layer, one reshaping layer, and three transposed convolutional layers. The transposed convolutional layers are used to up-sample the latent representation of images from the latent space. The kernel size of (3,3) is used in the layers of the decoder. The

strides equal to (2,2) are used to up-sample the input images to original dimensions. The final shape of the output from the decoder must be equal to the original input images, that is, 384×384. All the transposed convolutional layer uses the ReLU as activation function except the last or output layer of decoder uses Sigmoid as an activation function.

The custom loss is defined as the mean of KL divergence and reconstruction loss in terms of MSE or L2 loss and is represented by equation 5.1. The model is complied with custom loss, and Adam is used as an optimizer in the model.

$$VAE\,Loss\,Function = Reconstruction\,Loss + KL\,Divergence\,Loss \tag{5.1}$$

5.5 Threshold Selection and Classification Approach

Several methods are used to classify the outliers and unhealthy data from the healthy data in unsupervised learning based on the optimal threshold. In Machine Learning (ML), the threshold is the probability or some value in data that is used to classify the data and outliers or anomalies in the binary classification. It gives the trade-off between the false positives and false negatives in the data. The threshold is selected in several ways; most of them are based on how data is distributed and scaled in that distribution. Pete. R. Jones, in the paper [46], stated several methods to detect the outliers in psychological data; however, in this thesis, the method used to select an optimal threshold for classifying unhealthy images from healthy images is based on the Inter-Quartile Range (IQR) method. The IQR is the measure of the data variability about the median; that is, it represents the range in which half of the data is distributed. The Q_2 or median is the second quartile of the data in the distribution. It is essential to understand that the IQR method does not assume the normality of the distribution instead of other methods, which are based on the mean and standard deviation of the data and assumes the normal distribution of data. The equation 5.2 represents the mathematical calculation of IQR.

$$IQR = Q_3 - Q_1 \tag{5.2}$$

In equation 5.2, Q_3 is the third quartile of the data, representing 75% of the data lies between minimum and Q_3, whereas Q_1 being the first quartile represents the 25% of data lying between minimum and Q_1. A decision boundary is calculated for the binary classification so that any point in data that lies beyond that boundary is considered an

unhealthy data point or outlier. The equation 5.3 shows how the threshold range or decision boundary is selected in this thesis using the IQR method.

$$Optimal\,Threshold = Q_2 + (F \times IQR) \tag{5.3}$$

In the above equation 5.3, Q_2 is the median of the data points and F is the factor that controls the sensitivity of the threshold. F is based on how the data is distributed and maybe different for different DL models and error calculations. Therefore, the model classifies the data point as an outlier if it lies more than F times the IQR range from the median. One of the drawbacks of the IQR method is its insensitivity to the data that is not assumed to have a Gaussian or Normal distribution. This method also helps in the cases of detecting only the extreme outliers [46].

The threshold or decision boundary is based on the distribution of reconstruction loss in this thesis rather than probabilities of distribution as the binary classification will be performed based on either pixel-wise reconstruction error between the original and reconstructed images or SSIM score between the original and reconstructed images. The MSE score will be higher for the images with lesions as the model does not reconstruct the unhealthy portion of the image because of its training on only healthy images. Similarly, the SSIM score will also be lower for the images with a lesion than the healthy images. Based on the threshold, the data is classified using several ranges of thresholds. The unhealthy images are considered as positive values, and healthy images are negatives in this thesis. Therefore, keeping in mind the binary classification technique, the images are classified into total positives and total negatives. The classified values are validated against the already existing labels to calculate the true positives and negatives of the total positives and negatives. The false-positive rate, true positive rate, false-negative rate, and true negative rates are calculated to analyze the model performance and to construct the ROC curve from these rates. The AUC is also calculated to measure the model's performance and how efficiently the model can classify two different classes from the data. The overall accuracy, specificity, sensitivity, precision, and recall are also calculated on all the thresholds. The optimal threshold is selected to obtain the best possible classification from the models used in this thesis.

5.6 Lesion Detection

It is assumed that the trained model only reconstruct the healthy region of unhealthy images because the model is only trained on the healthy images (see section 5.2). This concept is used to detect the lesions in the unhealthy images by calculating the SSIM

score between the original and reconstructed images. The SSIM score will naturally decrease for the unhealthy image as the model does not reconstruct the unhealthy part of the image. Furthermore, the lesion or unhealthy region will show an even lower SSIM score than the whole unhealthy image, which will confirm the presence of a lesion in the image. The SSIM score of the unhealthy part of the reconstructed images is compared with the SSIM of the whole image, and the difference is displayed to get the visual inspection of whether the model can detect lesions or not.

Chapter 6

Experimental Evaluation and Results

In this chapter, the experiments, classification evaluation, and results using the proposed methodology are documented. The different configurations of test data-sets use to test proposed models are also explained in this chapter. This chapter does also include visual examples of the results produced.

6.1 Consistency of Variational Autoencoders

The thesis uses VAE and AE for the proposed methodology. The primary focus of the thesis lies on VAE due to the dynamic range and variability provided by VAE; however, AE is acting as a baseline model in this thesis, providing the basis of comparison to check whether the proposed method gives improved results when employed with VAE. Due to the randomization of initialized weights, it is imperative to assume that model will produce different results if run multiple times. To overcome this problem, the model is run with random seeds of 22, which means that randomization is optimized to produce the same set of numbers that corresponds to the random number sequence of 22 in the NumPy library.

6.2 Finding the best VAE Model

While training a VAE or AE, several parameters are to be tested to get the best possible configuration of the VAE. Some of these parameters were restricted to certain values due to the huge size of T2w data, which impose computational limitations for the system to carry a task. For this reason, latent dimension size could not be increased beyond 512 units, and batch size could not be increased beyond 128. The VAE is tested which

several configurations of parameters to try and find the best possible configuration. The size of latent dimension, batch size, learning rate, and the number of filters are varied to try and find the best parameters.

6.2.1 Learning Rate

The number of models is trained with a learning rate of 0.001 and 0.0001. The results of all the learning rates were recorded, and model reconstruction loss plots were generated to get a better understanding of the results. All the results obtained using different learning rates are documented in Appendix B. The learning rate of 0.0001 gives the best result in terms of reconstructed images and classification tasks and was chosen for model configuration. The model training and validation loss using a learning rate of 0.0001 for chosen VAE model while training for 1500 epochs is shown in figure 6.1.

6.2.2 Batch Size

The three batches size were used during the training of models, 32, 64, and 128. All the plots regarding different configurations of models with different batch sizes are documented in Appendix B. The difference in the results is not as distinct for 64 and 128 as for the learning rate case; however, there are still some minor differences. The batch size of 32 deemed the worse results. Even though the difference between the results of 64 and 128 is not huge, the batch size of 128 starts with a higher computational cost than for the batch size of 64. Also, the computational time of the model with batch size 64 is somewhat better than 128. Therefore, due to these minor differences, a batch size of 64 is selected. The model training and validation loss using a batch size of 64 units for chosen VAE model while training for 1500 epochs is shown in 6.1.

6.2.3 Number of filters

The higher number of filters gives out the higher number of abstractions that a network can extract from image data, resulting in large latent vectors. The larger the latent vector, the more information will be stored and available for the decoder to produce more precise and clear reconstructed images. However, it is not always true that the VAE, which produces the best-reconstructed images, will produce the best classification results. In this experiment, the following number of filters were used, 16,32, and 64. These, in combination with other VAE models, produced different sizes of latent vectors. The results from the different models using the different number of filters are present in

Appendix B. The interpretation of results encouraged using a combination of different filter sizes, that is, to use different filter sizes in different layers of the model.

6.2.4 Size of Latent Dimension

The size of the latent dimension is an essential feature in VAE. Even though it does not contribute directly to the model performance, it enhances the encoder's complexity and allows us to save more latent variables in the latent space. The three sizes of latent dimensions 128, 256, and 512 units are used for T2w data while 32 and 64 units for ADC due to lower dimensions. The plots generated using different models with different latent dimensions size are shown in Appendix B. The results show that there is not much difference in the reconstructed images obtained using different latent space sizes. However, latent space of 512 units increases 7 seconds of computational time per epoch size of 256 units increase the computational time to 3 secs per epoch. The classification results were better when the model with a latent space of 128 units was used as compared to the rest of the sizes for T2w and 64 units for ADC data. The model training and validation loss using latent space of 128 units for the chosen model while training for 1500 epochs is shown in figure 6.1.

6.2.5 Number of Epochs

The number of epochs contributes significantly to the performance of the model and reconstructions output by model. The proposed model was trained with different epochs ranging from 150 epochs to 1500 epochs. The model loss for 1500 epochs VAE model is shown in figure 6.1 and the remaining figures for model loss for different number of epochs are documented in Appendix B. The number of epochs required also depends upon the hyper parameters of the model. For instance, the model with small batch size and small latent dimension size sometimes needs a more significant epoch number to optimize the reconstruction loss. The proposed model with selected parameters of 64 batch size, a learning rate of 0.0001, and latent space of 128 units is also trained on T2w with different epochs; however, training this model for 1500 epochs gives the best results in terms of both predictions made on test data and model reconstruction loss. The model with a latent space of 64 units gives the best results for ADC data.

6.2.6 Selected Configuration for VAE

Different VAE models with different configurations and hyper parameters are trained in this thesis. In this thesis, the model is not only selected based on better reconstruction

quality of images but the prediction made by every model is also checked for lower reconstruction errors to get a better understanding of how trained models are contributing in the classification of images. The question was arising that whether the model with the lowest reconstruction loss and best-reconstructed image quality is producing good enough results when tested on the unseen test images. Therefore, all the trained models were tested on a test datasets and histograms were generated to get an overview of healthy and unhealthy datasets. These histograms visualize the reconstructed error (MSE or SSIM) of the images stratified by the unhealthy and healthy class. These histograms from different models and modalities are given in section 6.7 of this chapter and Appendix C, along with their respective threshold boundaries. Based on these reconstruction quality and reconstruction errors, the optimized model is selected with the following parameters shown in table 6.1 .

Parameters for VAE	**Value**
Batch Size	64
Optimizer	Adam (Learning Rate = 0.0001)
Latent Dimension	128 units for T2w
	64 units for ADC
Number of Epochs	1500

Table 6.1: Selected Hyper-parameters for VAE.

Furthermore, the training and validation loss for the chosen model with the selected hyper parameters are shown in figure 6.1.

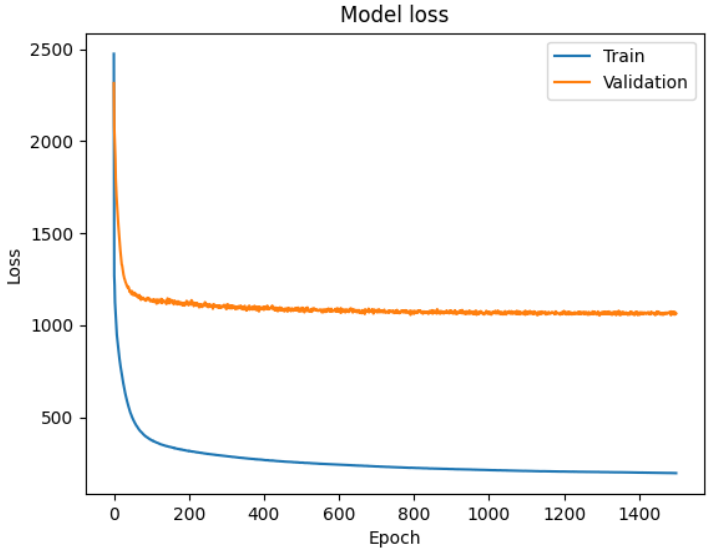


Figure 6.1: Training and validation loss for the selected VAE Model.

6.3 Finding the best AE

The same procedure is followed for finding the best AE as that of VAE. The number of configurations is experimented on by changing the batch size, learning rate, and the number of epochs. The different batch sizes and learning rates have not largely affected the AE, and reconstruction loss and image reconstructions remained somewhat mediocre. However, the number of epochs has some effect on the performance of AE and reconstruction ability. Therefore, AE is trained with batch sizes of 32, 64, and 128, learning rates of 0.001, and 0.0001 for 250, 350, and 500 epochs. Therefore, after analyzing the results from different models, the selected AE is trained with a batch size of 64, a learning rate of 0.0001, and training for 250 epochs gives the best results for the proposed model as a training model for more than 250 epochs tend to over fit the model and thus contributes poorly to predictions made on test images. Table 6.2 highlights the selected hyper parameters for the chosen AE model.

Parameters for VAE	**Value**
Batch Size	64
Optimizer	Adam (Learning Rate = 0.0001)
Number of Epochs	250

Table 6.2: Selected Hyper-parameters for AE.

6.4 Reconstructed Images from VAE versus AE

The trained models with selected hyper-parameters described in the section 6.2.6 and 6.3 of this chapter outputs the best-reconstructed images, and the quality of reconstructions is acceptable. The reconstructed images from VAE are better in quality than AE. Figure 6.2 shows the comparison of the reconstructed images for ADC images from VAE and AE. The VAE offers relatively realistic reconstructions of the images, and reconstructions from the VAE are somewhat closer to the original images. The image details are also prominent in the reconstructions made by VAE.

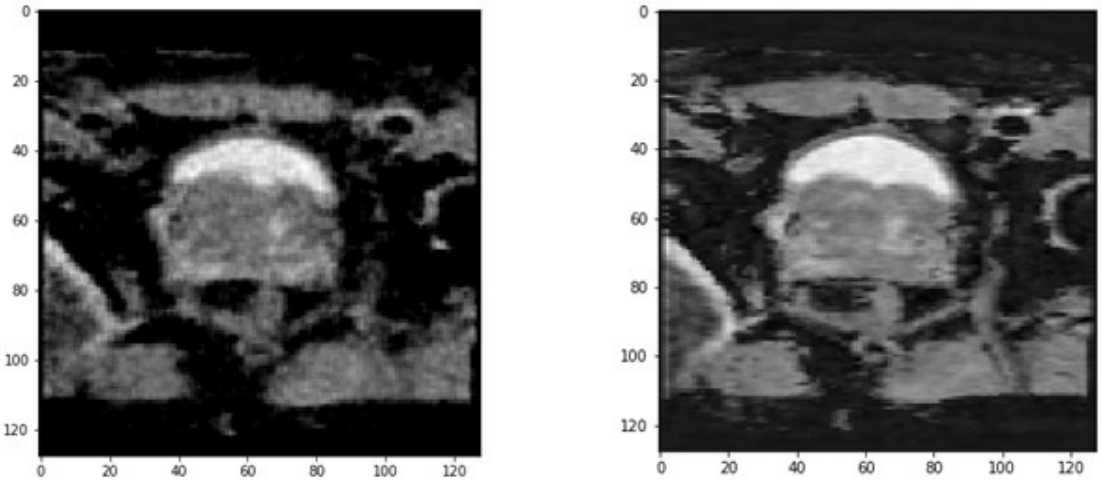


Figure 6.2: Illustration of reconstructions for ADC images from VAE (left) and AE (right).

Similarly, figure 6.3 shows the difference between the reconstructed images for T2w images from VAE and AE. It can be seen from figure 6.3 that VAE has better quality as compared to AE. The reconstruction from AE is blurrier for both T2w and ADC data as compared to VAE. The VAE model outputs reconstructions that look more consistent with the input in the healthy region.

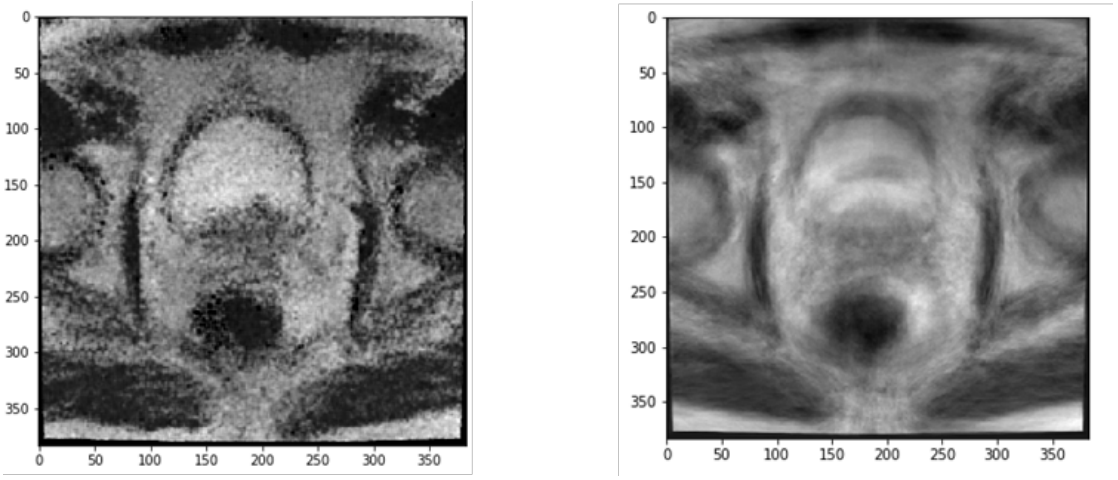


Figure 6.3: Illustration of reconstructions for T2w images from VAE (left) and AE (right).

The mean and standard deviation of reconstruction error for the healthy images is also higher for AE than VAE, as indicated by table 6.3 and confirms the better performance of VAE in image reconstruction.

DL Models	MSE for T2w		MSE for ADC	
	Mean	SD	Mean	SD
VAE	0.00259	0.00124	0.0046	0.00259
AE	0.00388	0.00151	0.00504	0.00527

Table 6.3: Comparison of MSE reconstruction errors of healthy images for VAE and AE.

6.5 Reconstructed Images for T2w versus ADC Images

The reconstructed images for ADC data have a better quality of reconstruction than the T2w data using VAE. Due to smaller dimensions and darker images, the model performed slightly better on ADC images. Figure 6.4 represents the comparison of reconstruction from T2w and ADC image data. It is evident from figure 6.4 that the reconstruction for ADC images is more precise and has better quality, in contrast to T2w, where reconstructions are somewhat blurrier and offer low-quality images.

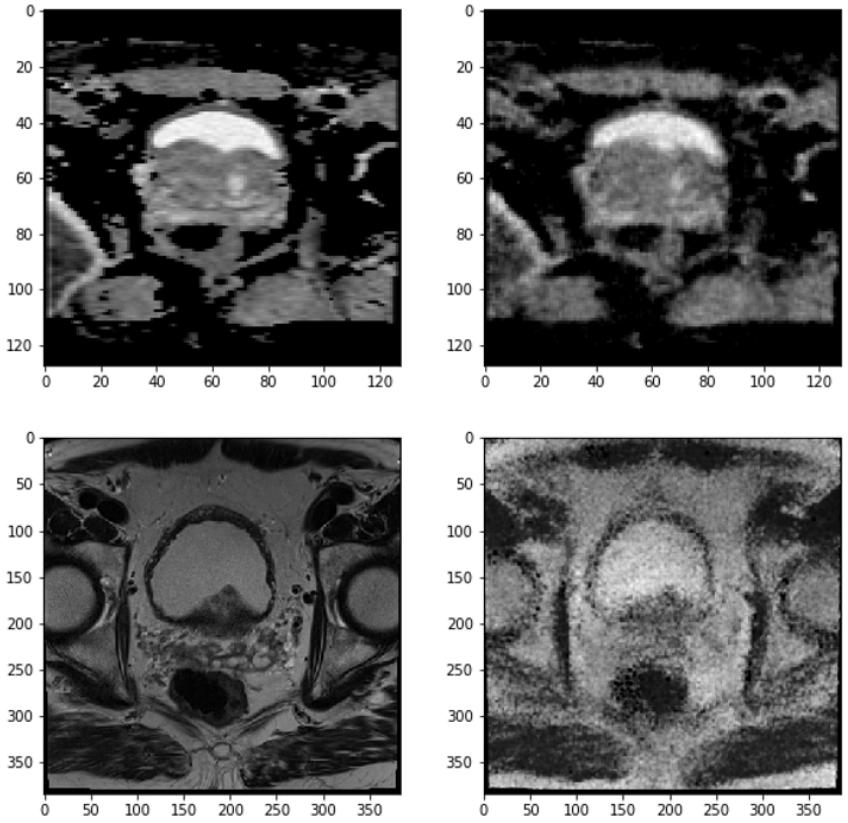


Figure 6.4: Illustration of ADC (first row) and T2w images (second row) with their respective reconstructions (original image is on the left and respective reconstructions are on the right of the figure.)

Table 6.4 shows the mean reconstruction error in terms of MSE and SSIM score in both modalities and provides evidence for the better performance of the model for ADC images. The MSE reconstruction error for healthy images in ADC is higher than that of T2w, thus contributing significantly better in the classification of two classes for ADC in contrast to T2w images where the difference between them is meager, offering a very narrow threshold range for T2w and decreasing the overall performance of the model. Similarly, the SSIM shows a similar trend for ADC and T2w, proving the superiority of ADC data over T2w.

Errors	**T2w**				**ADC**			
	Healthy		**Unhealthy**		**Healthy**		**Unhealthy**	
	Mean	**SD**	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**
MSE	0.00259	0.00124	0.00362	0.00219	0.0036	0.00258	0.00772	0.00258
SSIM	0.7523	0.0519	0.6557	0.0682	0.767	0.0861	0.5187	0.1646

Table 6.4: Comparison of reconstruction errors of healthy and unhealthy images for ADC and T2w data.

6.6 Reconstructions for Unhealthy versus Healthy Images

In general, the model behaves as expected in this thesis, and the reconstructions are acceptable for both classes of unhealthy and healthy images. However, comparing reconstructions for these two classes shows the noticeable difference for both modalities, as the difference can be evident in the table 6.4. The reconstructions of unhealthy images show higher MSE reconstruction loss or lower SSIM score as compared to reconstructions of healthy images, which provides the basis for the binary classification of two classes as the failure of the model to produce the lesion part of the unhealthy images provides the basis for higher MSE reconstruction error or lower SSIM score between the reconstructed and original image. The reconstructed images for the healthy images appear to be more precise, closer to the original images, and less blurry than unhealthy images. This trend is prominent for both modalities while using VAE for predictions.

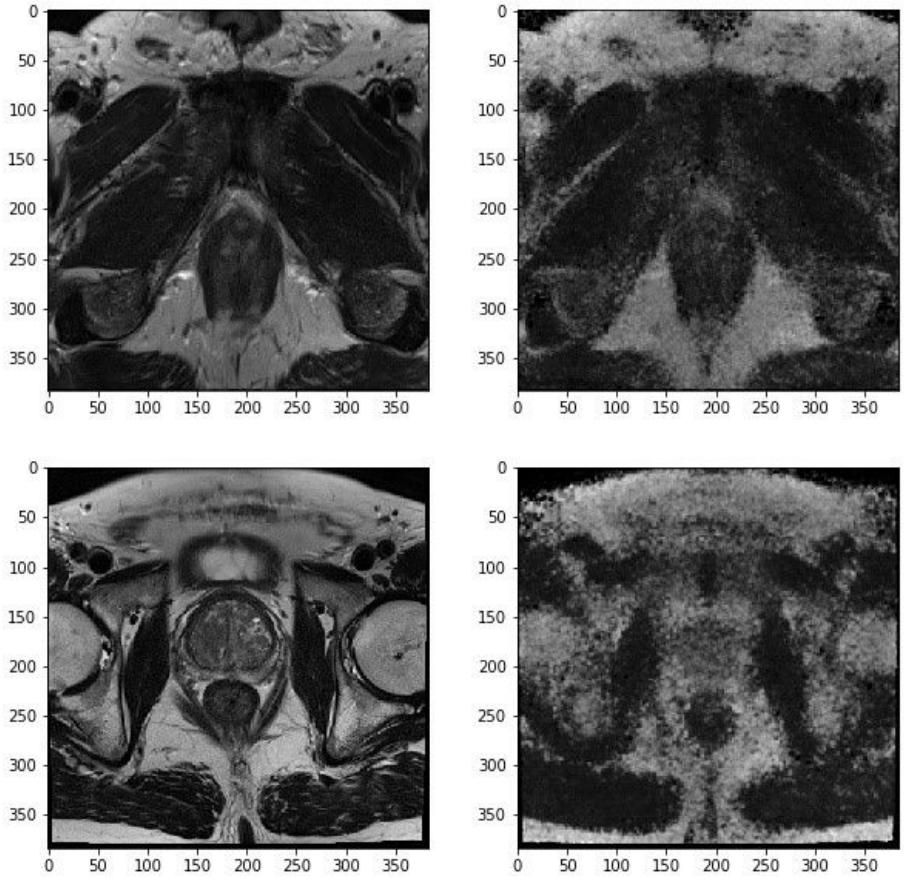


Figure 6.5: Illustration of Healthy (first row) and Unhealthy images (second row) with their respective reconstructions for T2w images (original image is on the left and respective reconstructions are on the right of the figure.

Figures 6.5 and 6.6 show the difference between the reconstructions for random unhealthy and healthy images from T2w and ADC images, respectively. It is evident from the figures 6.5 and 6.6 that the reconstructions for the healthy images have better quality and precise details than the unhealthy images, which tend to become blurrier. The interpretation of this difference between the reconstruction of two classes (unhealthy and healthy) indicates that the presence of lesions in the images negatively affect the reconstructions made by the model, which is only trained on the healthy images, thus helping in the classification of two classes and detection of lesions in healthy class.

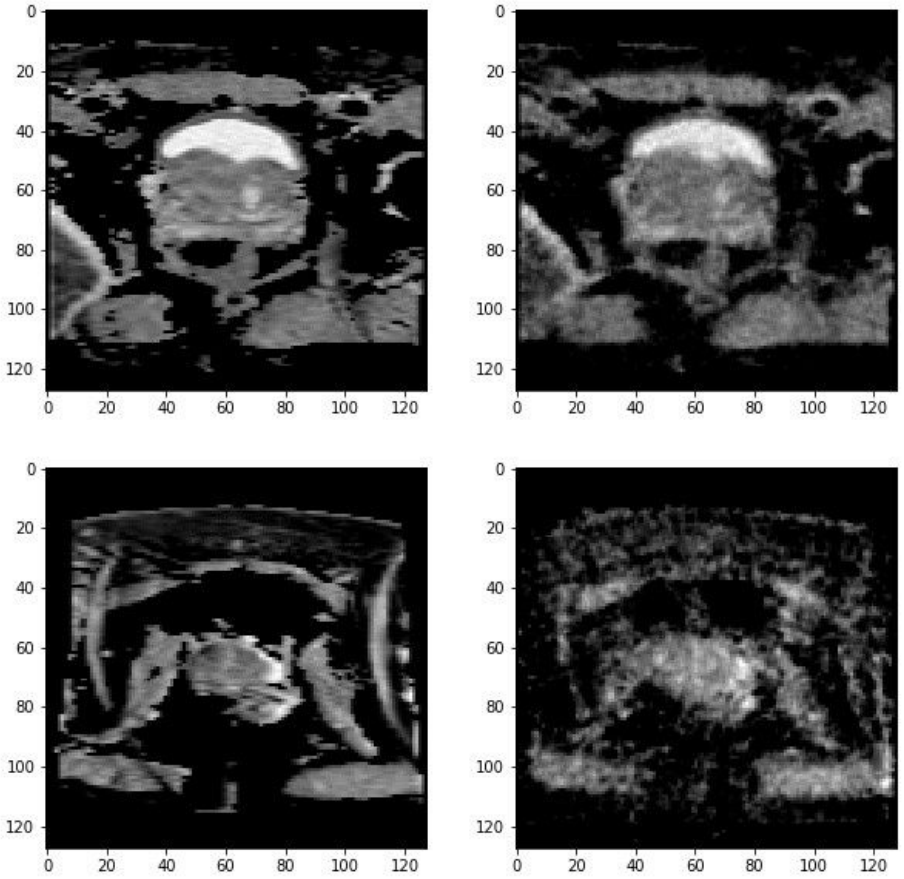
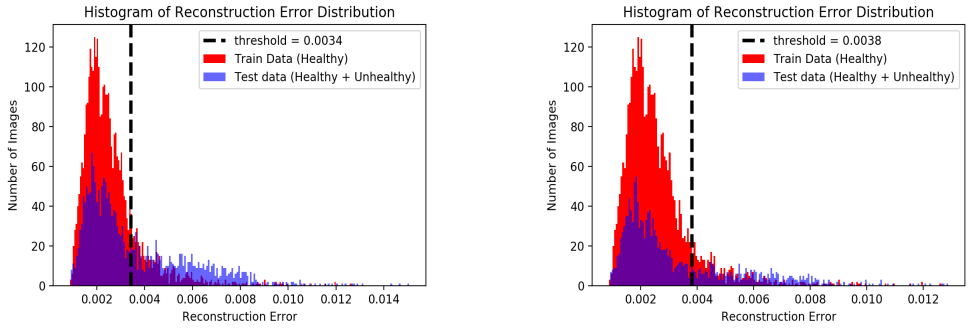


Figure 6.6: Illustration of Healthy (first row) and Unhealthy images (second row) with their respective reconstructions for ADC images (original image is on the left and respective reconstructions are on the right of the figure).

6.7 Threshold Selection

As explained in section 5.5 of the thesis, the threshold is selected using the distribution of pixel-wise reconstruction error or SSIM score between reconstructed and original images and the threshold is found by using the IQR method. The reconstructed image is classified as an outlier or unhealthy if it lies F times IQR range from the median of errors distribution. The optimal F values are selected to be in the range of [0,1.2] for MSE as reconstruction error, and [-0.8,0.4] for SSIM is used as a reconstruction error. These values of F give the best optimal range of threshold for the binary classification of two classes. Figure 6.7 shows the distribution of recorded MSE reconstruction errors for

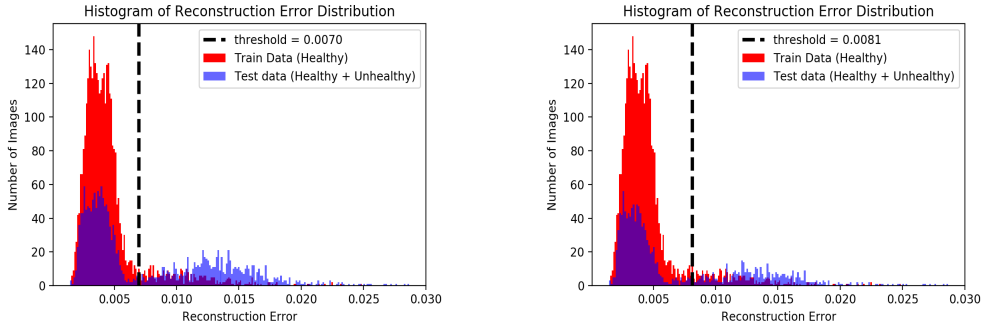
balanced and imbalanced dataset of T2w images along with their optimal classification boundary in an histogram while using VAE_{mse}.



(a) Balanced T2w data (b) Imbalanced T2w data

Figure 6.7: Histogram shows the distribution of MSE reconstruction errors for T2w images with optimal classification boundary.

Similarly, the figure 6.8 shows the distribution of recorded MSE reconstruction errors for balanced and imbalanced dataset of ADC images along with their optimal classification boundary in an histogram while using VAE_{mse} .



(a) Balanced ADC data (b) Imbalanced ADC data

Figure 6.8: Histogram shows the distribution of MSE reconstruction errors for ADC images with optimal classification boundary.

The remaining distribution histograms for other models and both modalities are documented in Appendix C.

6.8 Model Performance and Classification Results

In this chapter, the classifier's performance is tested against different configurations of the test dataset from both modalities, T2w and ADC. The section will also explain the

comparative results using different proposed DL models. The range of optimal threshold for classification is calculated using either pixel-wise reconstruction MSE/L2 or SSIM loss. (see section 5.5).

The classification is performed on the T2w and ADC test images. As mentioned in chapter 4, the test dataset contains both unhealthy and healthy images from the original dataset; however, none of these images are seen by the trained model, and all the slices of one subject lie in one dataset and are not repeated in another dataset (see section 4.2.4).

6.8.1 General Classification results

The classification results obtained by using VAE for predictions are far better than AE in terms of both classification accuracy and ROC- AUC. The ROC-AUC is calculated using the Sklearn library of Python and is considered a macro ROC-AUC score as a default setting. Table 6.5 shows the ROC-AUC and average classification accuracy for an optimal threshold for both modalities using VAE and AE with different reconstruction losses. Both MSE and SSIM are used for calculating reconstruction error between the reconstructions and original images. It is also important to understand here that the table 6.5 only gives the results for a balanced test dataset of two modalities.

Modalities Used	ROC-AUC				Classification Accuracy (%)			
	VAE_{mse}	VAE_{ssim}	AE_{mse}	AE_{ssim}	VAE_{mse}	VAE_{ssim}	AE_{mse}	AE_{ssim}
T2w	0.72	0.72	0.64	0.53	78.7	80.3	70.5	53.6
ADC	0.81	0.76	0.74	0.51	81.9	81.7	81.7	59.7

Table 6.5: Comparison of ROC-AUC and Classification Accuracy of all models for balanced T2w and ADC data.

From table 6.5, it is evident that the VAE model performed significantly better than AE. Moreover, the classification was slightly better when reconstructions are made for ADC using VAE than AE. The same results can also be seen from the figures 6.9 and 6.10, showing the ROC curve for both modalities using VAE and AE. Note that the results in the figures 6.9 and 6.10 are also only documented for the balanced test data of both modalities.

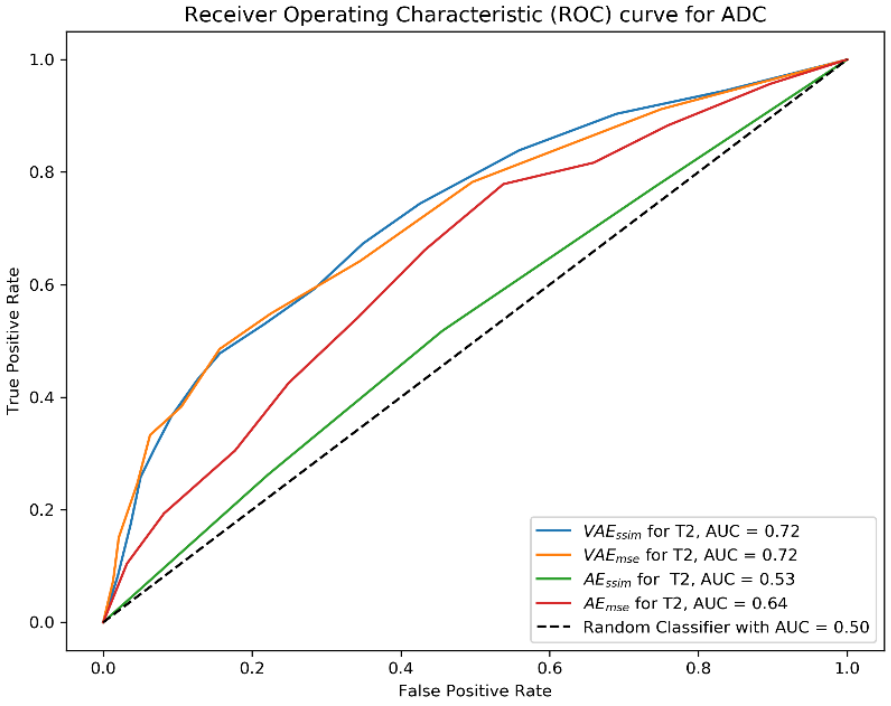


Figure 6.9: Illustrates ROC-AUC curves of all the models for balanced T2w data.

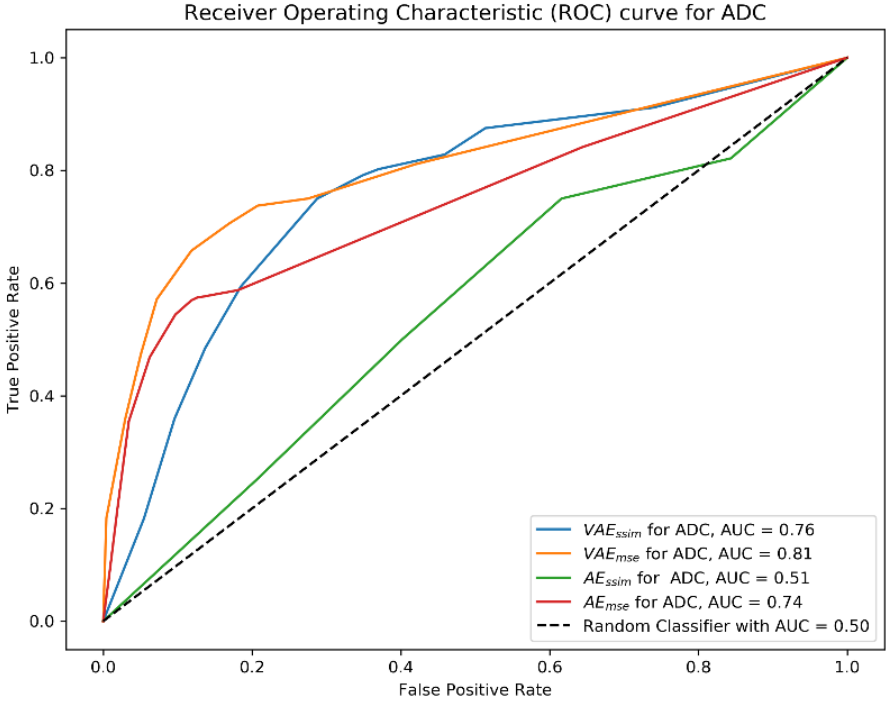


Figure 6.10: Illustrates ROC-AUC curves of all the models for balanced ADC data.

Moreover, even though the VAE_{ssim} shows a higher average sensitivity of approx. **0.79** for T2w as compared to VAE_{mse}, AE_{mse}, and AE_{ssim}, which shows the average sensitivity of **0.71**, **0.60** and **0.49** for T2w respectively on the range of selected threshold, yet the consistency of VAE_{mse} is higher and does not change abruptly compared to other models. The precision of VAE_{mse} is also better as compared to other models. SSIM is believed to have a better distinction property for different distortions in an image and can help to detect the lesion in MR images by measuring the similarity between two images. The AE_{mse} performed marginally better for both modalities as compared to the AE_{ssim} which produce poorly unacceptable results. $AE_{ssim}$is not consistent with the results and shows a considerable bias towards the healthy images. All the models significantly performed better in classifying the ADC data set than the T2w images, which can be deduced from the table 6.6, that the respective recall and precision values of two modalities for all proposed models. Therefore, the presented results suggested somewhat comparable results for both VAE_{mse} and VAE_{ssim}, and the model's performance is recorded to be equally acceptable for both configurations of VAE compared to both AE models, which gives poor results of image classification.

Modalities Used	Recall				Precision			
	VAE_{mse}	VAE_{ssim}	AE_{mse}	AE_{ssim}	VAE_{mse}	VAE_{ssim}	AE_{mse}	AE_{ssim}
T2w	0.71	0.82	0.66	0.62	0.86	0.78	0.76	0.55
ADC	0.72	0.71	0.71	0.61	0.92	0.93	0.91	0.50

Table 6.6: Comparison of Recall and Precision of all models for balanced T2w and ADC data.

The other relevant figures related to classification accuracy, Precision-Recall curves, and other related statistical model performance parameters are documented in Appendix D.

6.8.2 Classification Results for Different Test Data Configurations

The VAE is tested on the two configurations of test data, the balanced and imbalanced test data using both modalities, and the results are analyzed to obtain the model's performance, whereas AE is only tested on the balanced test dataset. The balanced test data for T2w contain an equal amount of healthy and unhealthy images, that is, 50% distribution for both classes that sums up to give 1033 images for each class. However, imbalanced data contain 1033 healthy and 500 unhealthy images for T2w MRIs. Similarly, the ADC MRI balanced test data also contains approximately equal images of healthy (907) and unhealthy data (1028). The imbalanced test data for ADC contains 907 healthy images and 514 images of unhealthy images. The experiments were performed on both the data configurations, and the results are documented with explanation.

Modalities	Configuration	ROC-AUC		Classification Accuracy (%)	
		$\mathbf{VAE_{mse}}$	$\mathbf{VAE_{ssim}}$	$\mathbf{VAE_{mse}}$	$\mathbf{VAE_{ssim}}$
T2w	**Balanced Test Data**	0.72	0.72	78.7	80.3
	Imbalanced Test Data	0.76	0.78	85.8	87.2
ADC	**Balanced Test Data**	0.81	0.76	81.9	81.7
	Imbalanced Test Data	0.9	0.83	90.3	90.1

Table 6.7: Comparison of ROC-AUC and Classification Accuracy of all VAE models for different configurations of T2w and ADC data (balanced and imbalanced data).

The table 6.7, the figures 6.11 and 6.12 show the results of experiments performed on balanced and imbalanced test data for both modalities of T2w and ADC. It is evident from the table that imbalanced test data tend to provide better classification results with higher ROC-AUC and higher classification accuracy in contrast to the balanced dataset. For $\mathbf{VAE_{mse}}$ with **T2w imbalanced** test data, an average ROC-AUC of **0.76** is recorded as compared to the **balanced** test data, which shows the ROC-AUC of **0.72**. However, the difference is more prominent for $\mathbf{VAE_{mse}}$ with **ADC** test datasets where **imbalanced** and **balanced** test sets show the ROC-AUC of **0.90** and **0.81**, respectively. Similar behavior is shown by the VAE_{ssim} for balanced and imbalanced datasets. The model's sensitivity is also recorded to show an increase when the model is tested on imbalanced test data compared to the balanced test dataset.

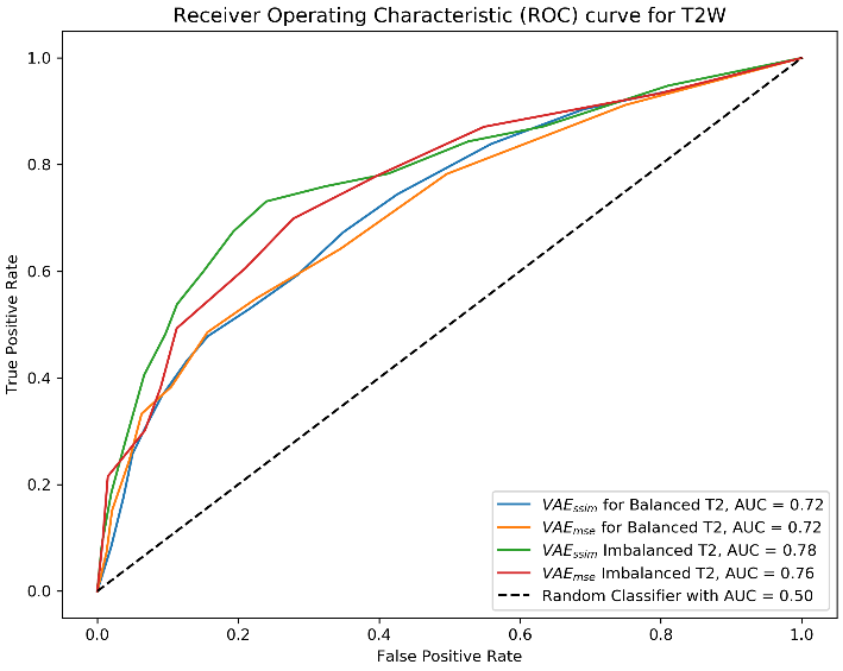


Figure 6.11: Illustrates ROC-AUC curves for the two configurations of T2w data, balanced and imbalanced.

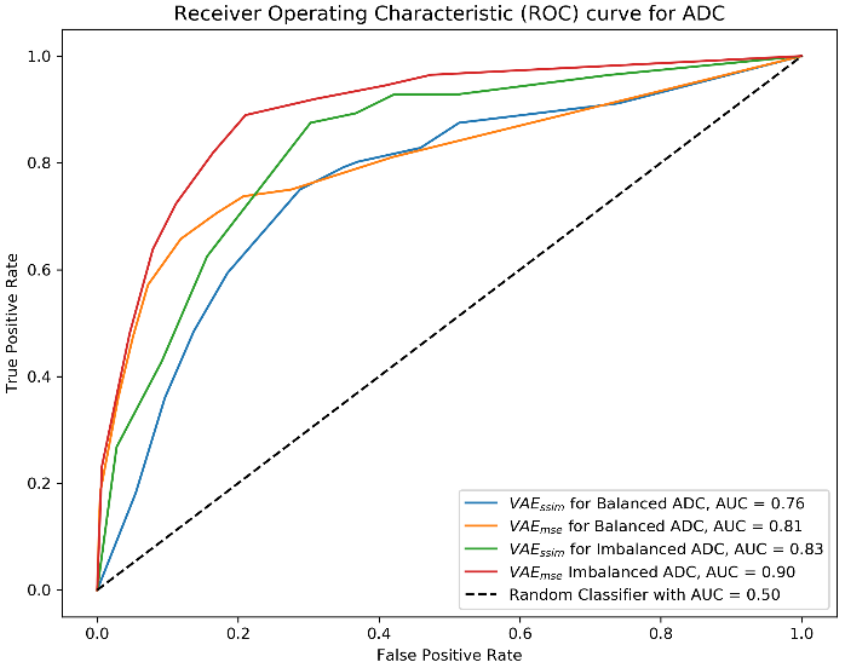


Figure 6.12: Illustrates ROC-AUC curves for the two configurations of ADC data, balanced and imbalanced.

The confusion matrix for the ADC and T2w images using VAE_{mse} are presented in the figures 6.13 and 6.14 for balanced and imbalanced test data to get the general overview of the classification of two classes in the dataset. The corresponding confusion matrix of AE_{mse}, AE_{ssim} for balanced T2w and ADC data and confusion matrix of VAE_{ssim} for both configuration of test data of ADC and T2w images are presented in Appendix D.

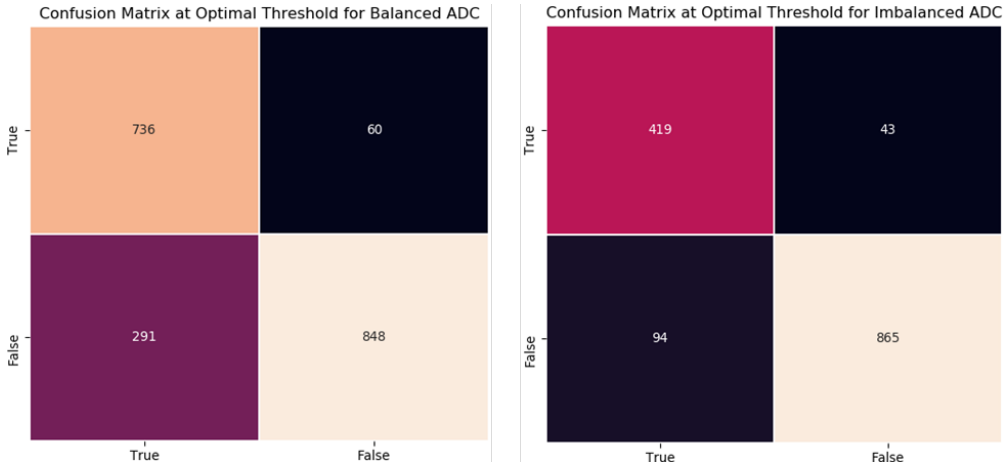


Figure 6.13: Shows the confusion matrix for the two configurations of ADC data, balanced (left) and imbalanced (imbalanced) while using VAEmse. The unhealthy images are considered as positives and healthy images are considered negatives.

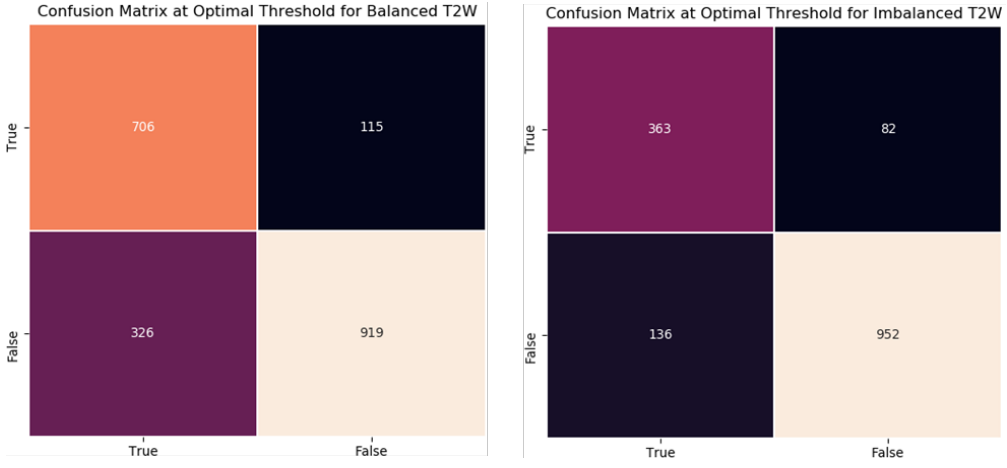


Figure 6.14: Shows the confusion matrix for the two configurations of T2w data, balanced (left) and imbalanced (imbalanced) while using VAEmse. The unhealthy images are considered as positives and healthy images are considered negatives.

The analysis of figures 6.13 and 6.14 indicates the increased number of true positives (actual healthy images) and true negatives (actual unhealthy images) for imbalanced dataset compared to balanced one. Thus, the class imbalance surely increases the performance of the model, and the model classification ability to distinguish healthy and unhealthy images improved by imposing the class imbalance in the test dataset.

6.9 Validation of Lesion detection in Reconstructions

As explained earlier in chapter 5, the basic concept of the thesis is to reconstruct only the healthy region of the unhealthy images by using the model that is only trained on the healthy images, leaving behind the lesions, not to be reconstructed from the original images. The results explained in section 6.6 of this chapter suggested that the model does not reconstruct the lesions as the reconstructions from the unhealthy images have lower SSIM scores than healthy images.

The figures 6.15 and 6.16 show two random T2w unhealthy images with their respective lesion masks, their reconstructions, and the difference between the original and reconstructed images with the masks of unreconstructed regions in the original image. Line-wise, the first image is original unhealthy image (left), second image is its respective reconstructions (center) and third image is the difference image with mask of unreconstructed region on original unhealthy image (right) in the figures 6.15 and 6.16. The white box highlights the lesion region in the images. The highlighted red region in the image represents the regions of the reconstructed image where the model does not fully reconstruct the features present in the original image, or the reconstructed region shows the structural difference compared to the original image. The un-highlighted parts of the difference image show the regions, which gives the structural difference of none to almost zero in reconstructions compared to the original image, and reconstruction of the image was perfect at those regions. The images are documented for VAE_{mse} on T2w unhealthy images.

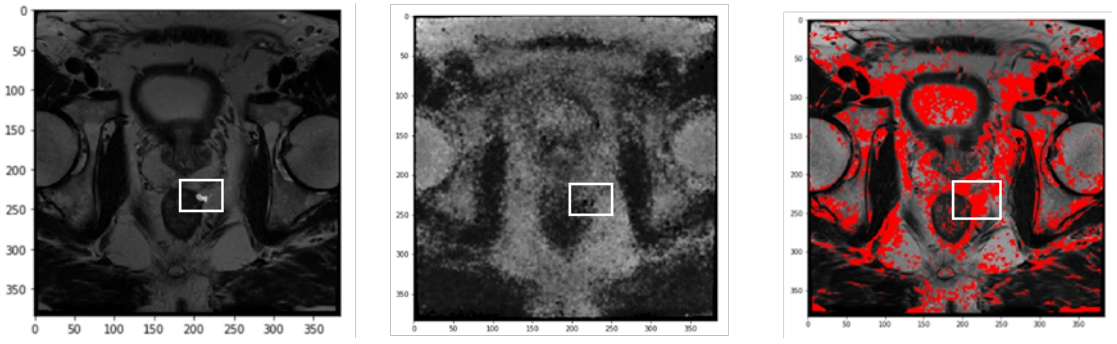


Figure 6.15: Image 1 - Illustration of random unhealthy T2w image with lesion mask (left), its respective reconstructions (center) and the mask (red region) of unreconstructed region on original image (right). The lesion region in all the images is highlighted by white box.

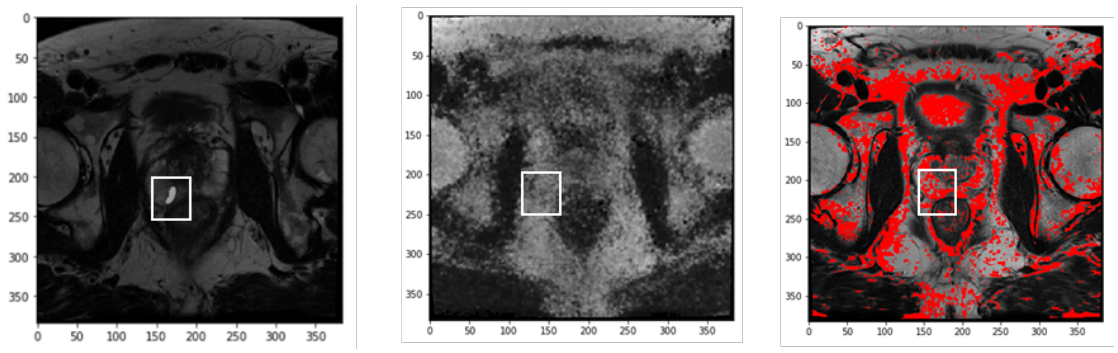


Figure 6.16: Image 2 - Illustration of the unhealthy T2w image with lesion mask (left), its respective reconstructions (center) and the mask (red region) of unreconstructed region on original image (right). The lesion region in all the images is highlighted by white box.

Even though the reconstructions made by the model are a bit blurry, the model can successfully detect the anomalies in the images, and only the healthy features of the images are reconstructed. It is evident from the images in figures 6.15 and 6.16 that the model does not reconstruct the lesion region of the unhealthy image and comply with the original proposed concept of lesion detection (see chapter 5).

Another validation technique is to measure SSIM for the lesion region of the reconstruction images and compared it with the overall SSIM of the whole image. As the lesion is not reconstructed in the reconstructed images, the lesion region will offer an even lower SSIM score compared to the overall image. The table 6.8 shows the SSIM score of the lesion region of two reconstructed images in comparison to the respective overall SSIM of whole reconstructed images (center images in figures 6.15 and 6.16).

Images	Lesion Region	Whole Image
Reconstructed Image 1	0.423	0.658
Reconstructed Image 2	0.3722	0.605

Table 6.8: Comparison of SSIM score for lesion region and whole image for random T2w unhealthy images.

It is evident from the table 6.8 that the lesion region of the reconstructed image shows a lower SSIM score as compared to the overall reconstructed image, which confirms the presence of a lesion in that region and supports the concept presented in chapter 5 of this thesis. The SSIM score changes abruptly when the size of the lesion increases in the MRIs, as shown in table 6.8 , in which image 2 has a bigger lesion size compared to image 1, lowering the SSIM score even more for image 2.

Chapter 7

Discussion

This chapter presents the discussion of the achieved results, factors that affect the performance of proposed models and binary classification of images, limitations of the thesis, and comparison of the results presented in this thesis to the related scientific works.

7.1 Effectiveness of Proposed Methodology

The analysis of achieved results presented in chapter 6 of this thesis indicates that the proposed method successfully classifies the prostate MR images into two classes of unhealthy and healthy images and can detect the lesion in the images. It provides a significantly simple yet effective unsupervised approach for lesion detection and estimation of outliers by learning the probability distribution of healthy images. The average ROC-AUC of **0.80** and classification accuracy of **83**% for the proposed models provides solid evidence that the models can achieve a human-level lesion detection without using any labeled data. The flexibility of the proposed method allows it to work with any complex image dataset, confirmed by the fact that the model is tested on datasets from two modalities (T2w and ADC). Even though the direct comparison between the studies can be difficult, since different data sets are used, the results produced in the thesis for the proposed methodology are justifiably admissible for comparative studies.

7.2 Evaluation of DL Models

Each of the presented models shows an ability to detect the lesions; however, their performance differs. The achieved results presented in the chapter 6 of this thesis

indicates the superiority of variational autoencoders (VAE) over simple autoencoders (AE) in the task of lesion detection in prostate MRI. Though the reconstructions are a bit blurry, VAE is more consistent and retains the realistic structure of MRIs in reconstruction. The better performance of the VAE is related to the fact that their output is conditioned to the probability distribution of the latent vectors in compressed dimensionality, which helps VAE to distinguish the outliers in the probability distribution. However, it should not be assumed that AE cannot be used for the anomaly detection task but depends upon the complexity of the dataset and proposed methodology. The binary of classification of prostate MRIs by learning probability distribution of healthy images, does demand stochastic, which makes VAE a better DL model for this thesis as compared to the AE.

7.3 Impact of Data on Model Performance

The proposed model behaved differently for different modalities of prostate MR images used in this thesis, and the performance of the model noticeably varies for T2w and ADC images. Apparently, the model produces remarkably better results for ADC images with higher ROC-AUC, higher classification accuracy, and improved precision values for ADC images. The significantly improved model performance might be reasoned by the fact that the prostate lesions are mainly present in the peripheral zone (PZ) of the prostate, and only 20-30% of the lesions are present in the transition zone (TZ). The effective MRI sequence to detect the lesion that is present in PZ is DWI or ADC [47]. Therefore, ADC images offer better details about the presence of lesions in prostate MRIs and the model learns relatively better about the distribution of lesions in unhealthy images when ADC images are used. The small dimensional size of ADC images might be another noticeable factor that can improve the model's performance as the model can learn efficiently fast on smaller images.

7.4 Impact of Class Imbalance on Model Performance

The class imbalance also projects a significant impact on the performance of the model as the imbalanced test data noticeably improve the performance of the proposed models for both modalities as compared to balanced test data. It is believed that the class imbalance causes the machine learning model to develop a certain bias towards the majority class, thus subjecting the proposed model to frequency bias, in which the model is more likely to place emphasis on learning from the probability distribution of healthy images and pay close attention to the distribution of reconstruction errors for

healthy images. As explained in the Chapter 5, the proposed model is only expected to reconstruct the healthy region of an unhealthy image, thus increasing the reconstruction error for unhealthy images, which provides the basis for classification. Therefore, this biased behavior of the model might be eventually helpful in this thesis, enforcing the model to classifies every image as unhealthy whose reconstruction error even slightly differs from the normal distribution of reconstruction errors for healthy images.

It is imperative to understand here that the thesis uses the macro aspect of ROC-AUC for both configurations, which is mainly considered for the imbalanced datasets. The results for balanced datasets might differ if a micro aspect of ROC-AUC is used. However, in real-world problems, like a hospital, the number of healthier patients are way more than unhealthy patient, and imbalanced datasets can provide better insight to the classification of classes (healthy and unhealthy) as it gives a more realistic and practical approach to tackle class imbalance in these situations.

7.5 Limitations

This section will discuss some limitations of this project.

7.5.1 Dataset

Though the model was able to produce acceptable reconstructions, the larger size of T2w might have influenced the model's performance. The larger size of T2w images also made it very difficult to find the correct hyper-parameters and configurations for the model to train on T2w images. The training of the model and the classification of two classes for T2w images might have improved if the reconstructions were slightly better. It might have enhanced model performance if T2w images were down-sampled to the size of 256×256 pixels instead of up-sampling to 384×384 pixels.

7.5.2 Pre-Processing

There was no manual inspection conducted for all the images after the normalization of the images. Only a few random images are inspected from a dataset. This problem might have led to some unexpected issues for some subjects that might have influenced the results.

7.5.3 Computational Limitations

The training of these deep learning models like VAEs and GANs is time-consuming and computationally expensive. The training of the model required a computationally rich GPUs system. Though UiS GPU servers provided an excellent platform for working with these models, however, limited availability of these GPU servers and large queues to access the severs imposed certain constraints for training purposes. The larger size of T2w images also raises questions of having enough computational power to train model and classification experiments. Even with the powerful setups, the processing of these larger T2w images is time-consuming. During the training and classification of T2w images, the system seemed hesitant to perform certain operations due to memory issues, causing the server to crash many times. The batch sizes and size of latent dimensions could not be increased after a limited value for T2w images because of computational limitations as the server was sometimes unable to process computationally expensive T2w data.

7.6 Comparison to Related Work

To the best of my knowledge, there is no published work related to the unsupervised lesion detection in prostate images using VAE or AE on PROSTATEx challenge data.

The comparison of results with the methodology formulated in [11], [12], [13], and [14] suggested that the proposed model in the thesis can classify and detect the lesions in the MR images with acceptable model performance. The complexity of the dataset in this thesis offers specific challenges to the model and makes it difficult to achieve the same quality of reconstructions and model performance presented in [12] and [13]. However, their model results might be slightly misleading as they are based on the simple dataset, MNIST, KDD CUP 99, requiring minimal effort to get good model performance for these datasets.

The paper [11] does inspire the methodology of this thesis in the essence that both propose unsupervised reconstruction-based lesion detection in images; however, their focus was to address the lack of consistency in latent space representations and improve the model performance by adding constraints to the latent space representation. The VAE in their paper also produces somewhat blurry reconstructions; however, it does not hinder the ability of the model to detect lesions, as presented by this thesis. The main models used in their work include AEE and VAE with added constraints, in contrast to this thesis, where VAE and AE are employed for the task. The constraints on the

model do improve their model performance, and they manage to produce good results compared to the ones presented in this thesis.

The paper [14] addresses the problem of proper latent dimensionality for the data and introduces VQ-VAE as the reconstruction model to obtain a discrete latent representation of normal data and then employs PixelSail, an autoregressive model, to determine the parts that deviate from the normal distribution in the input latent space. This behavior helps them to get rid of the unwanted reconstructions of unhealthy parts. The model works with MVTec AD inspection images, consisting of 15 categories. Their results show the improvement of the model by 15% in terms of ROC-AUC. Even though their model proposed discrete latent representation, yet the results produced by them was somewhat comparable with the result presented in this thesis, especially for ADC images, where the average ROC-AUC (0.80) for the VAE model of this thesis is very close to the average ROC-AUC (0.86) of VQ-VAE model presented in their work. As mentioned earlier, the main reason for comparing other's work with the current thesis is not to prove the dominance of one's method over others, but to get the general idea of the results presented in this thesis compared to others.

Chapter 8

Conclusion and Recommendations

In this chapter, a conclusion of this thesis and the recommendations for future work is given.

8.1 Conclusion

This thesis proposed a new method of detecting lesions in prostate MR images in an unsupervised framework by learning the distribution of only healthy images while training and detect lesions in unhealthy images according to the healthy data distribution. The thesis employs two DL models, namely VAE and AE, to work with the two modalities of prostate MR images, T2w and ADC. The prostate MR images are taken from the PROSTATEx challenge data.

The thesis uses a two-step methodology where the first step involves the pre-processing of data and the training of proposed models and the second step classification of reconstructions made by trained model on test data, the optimal threshold selection and detection of lesions are performed in the second step. The thesis selects the optimal threshold based on the reconstruction error using the IQR method, and classification is done based on the pixel-wise reconstruction error between the reconstructions and the original images. The thesis directly compares the SSIM score of the lesion region to that of the whole reconstructed image.

A set of experiments are conducted to find the optimal set of parameters and structure for the proposed model. The results produced by the thesis suggest that the model successfully classifies healthy and unhealthy images. Generally, VAE performed better than AE for both modalities, and the model performance was better for ADC images

compared to T2w. The proposed model can detect lesions from the unhealthy images as the lesion region shows higher SSIM scores than the rest of the image.

To conclude, the obtained results support the primary aim of this thesis; that is, the proposed model can be employed to attain the human-level lesion detection ability by learning the distribution of healthy samples.

8.2 Future Recommendations

VAE has significantly become a popular method with DL, especially in the areas of anomaly detection and outlier estimation from the normal data. The training of VAE using the T2w images from PROSTATEx Challenge data has proved to be quite challenging as the dataset contained the larger images due to the up-sampling technique used in this thesis. Therefore, future work could involve the down-sampling of T2w images to provide experimental flexibility to work with the T2w images. The model performance could be enhanced after the down-sampling of T2w images as the model might produce better reconstructions of down-sampled T2w images.

It is always better to have more data for the model to work with as it could efficiently learn the distribution of data during training, and classification tasks might improve. The augmentation of a dataset can be considered for future work as it could be an effective technique to improve the overall model performance.

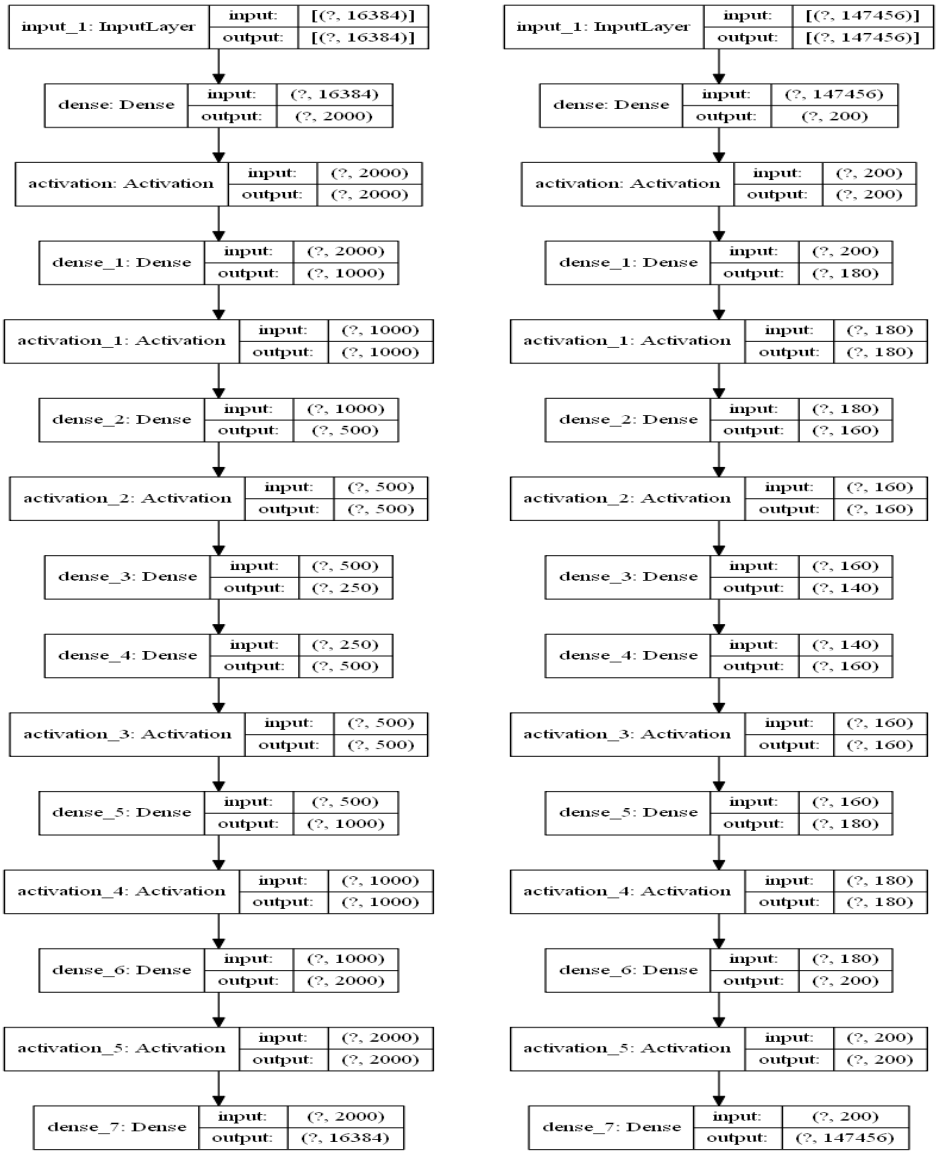
The regularization technique, addition of extra constraints to the latent space representation, as mentioned in the paper [11], can also enhance the model performance and can be considered as a potential future work. The use of VQ-VAE for discrete latent representation of normal data might improve the model performance in detecting the lesion from the unhealthy images. In contrast to VAE's indeterministic latent representation, the VQ-VAE has the ability to encode inputs into latent space with deterministic mapping and reconstructs data from quantized vectors. Therefore, VQ-VAE might be able to detect lesions from unhealthy images more efficiently. VQ-VAE could be employed as a comparative model for lesion detection using the proposed methodology as a future recommendation.

List of Figures

List of Tables

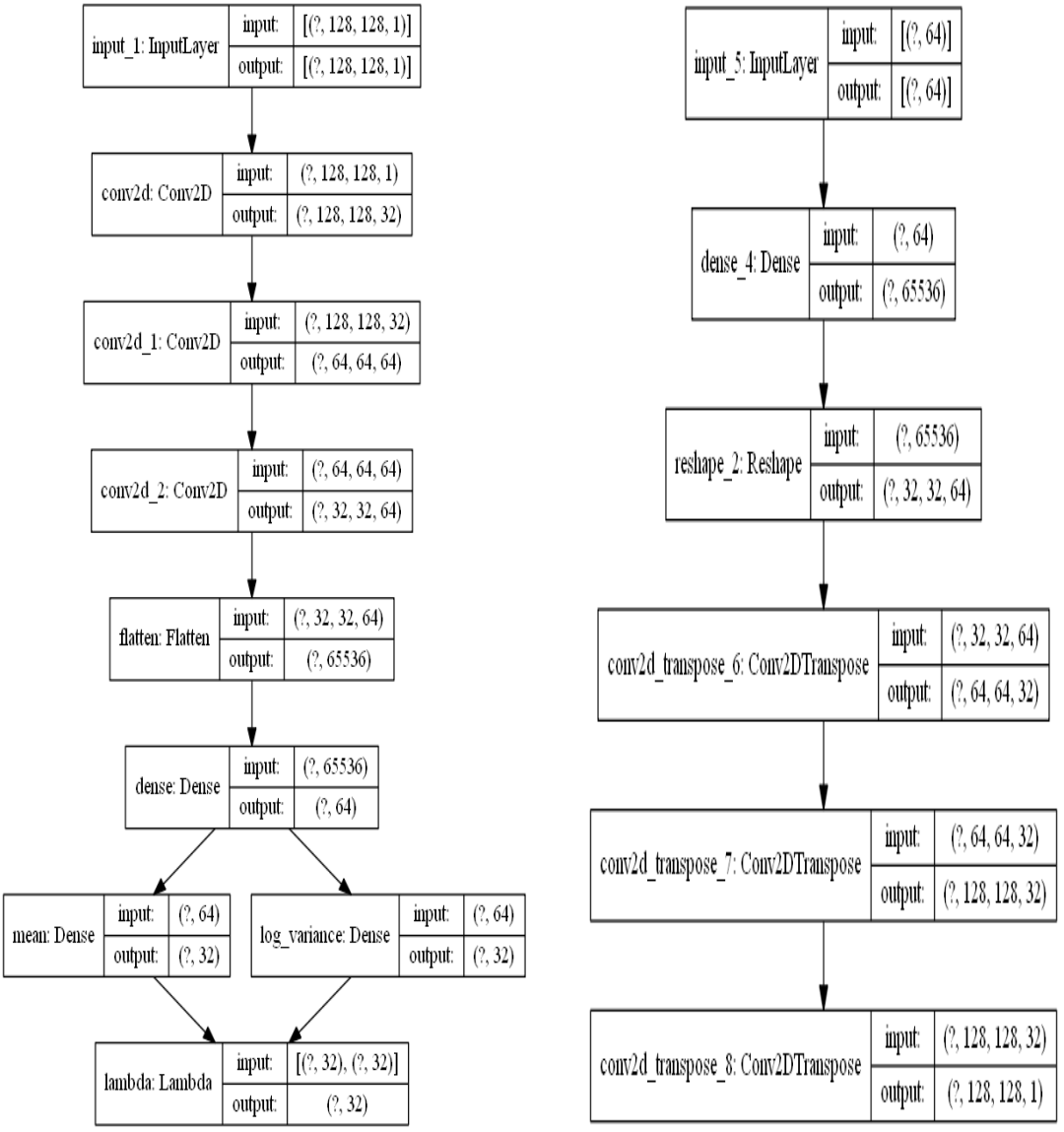
Appendix A

VAE and AE Model Design



(a) For ADC data. (b) For T2w data.

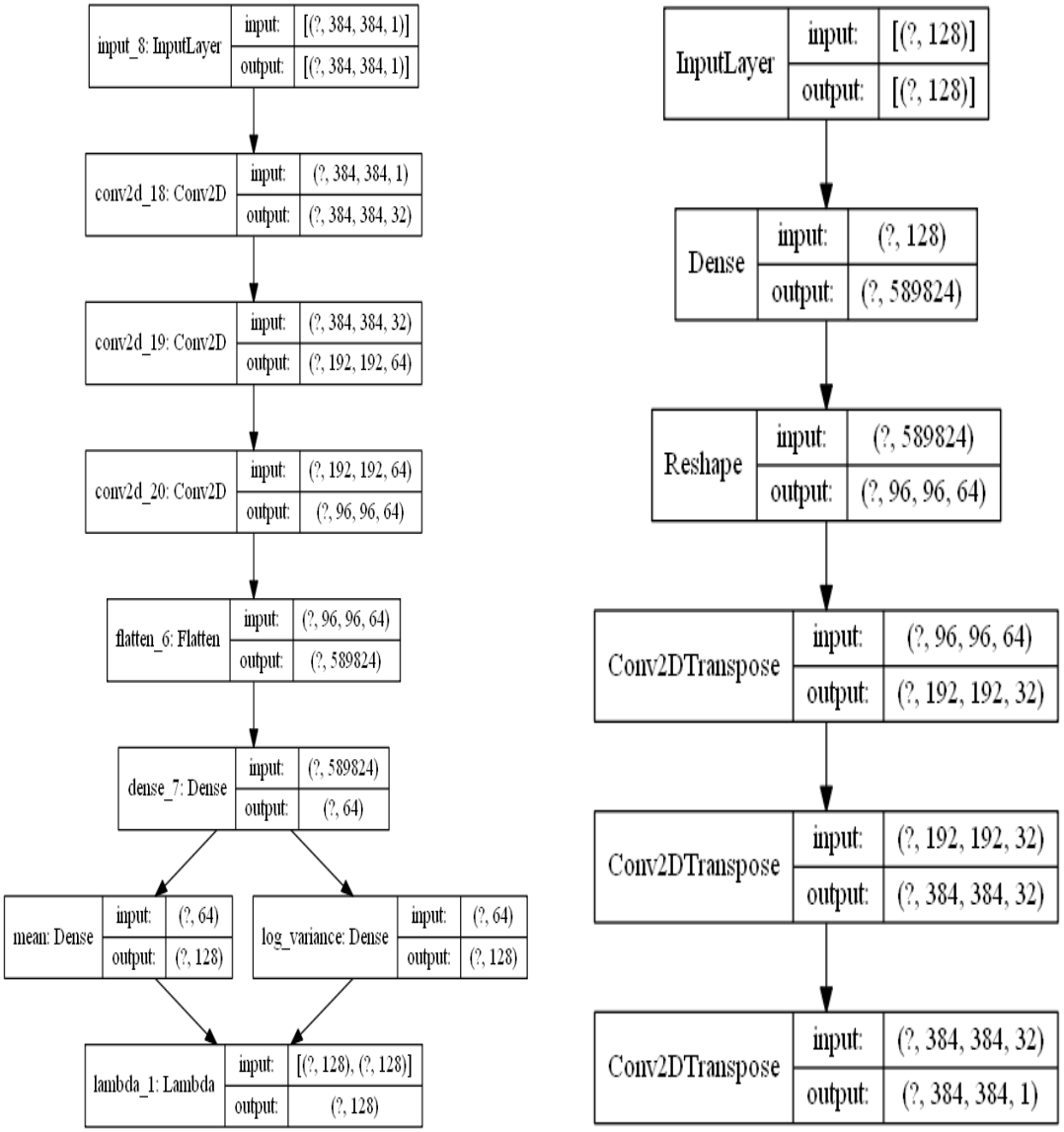
Figure A.1: AE Model Design for both modalities (T2w and ADC).



(a) VAE Encoder.

(b) VAE Decoder.

Figure A.2: VAE Model Design for ADC data.



(a) VAE Encoder. (b) VAE Decoder.

Figure A.3: VAE Model Design for T2w data.

Appendix B

Training and Validation Loss for VAE

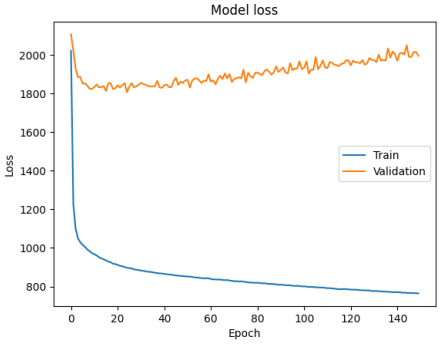


Figure B.1: 150 Epochs - VAE with LR = 0.0001, BS = 32, and LD = 128 units

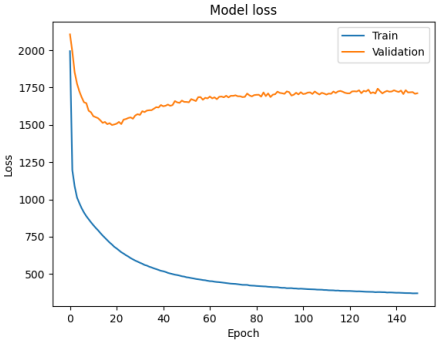


Figure B.2: 150 Epochs - VAE with LR = 0.0001, BS = 64, and LD = 256 units

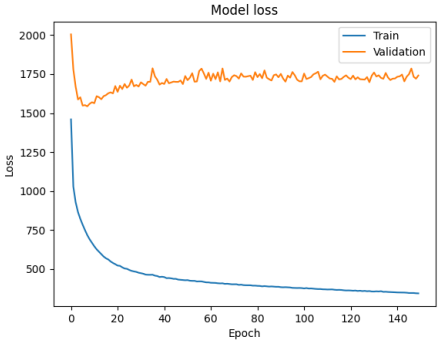


Figure B.3: 150 Epochs - VAE with LR = 0.001, BS = 64, and LD = 256 units

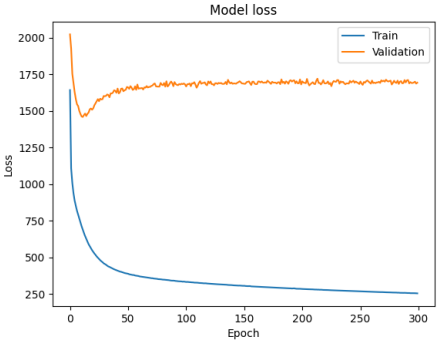


Figure B.4: 300 Epochs - VAE with LR = 0.0001, BS = 32, and LD = 128 units

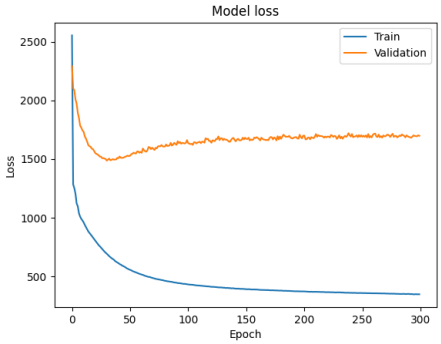


Figure B.5: 300 Epochs - VAE with LR = 0.0001, BS = 128, and LD = 512 units

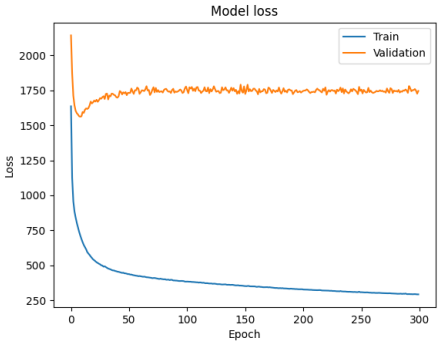


Figure B.6: 300 Epochs - VAE with LR = 0.001, BS = 64, and LD = 256 units

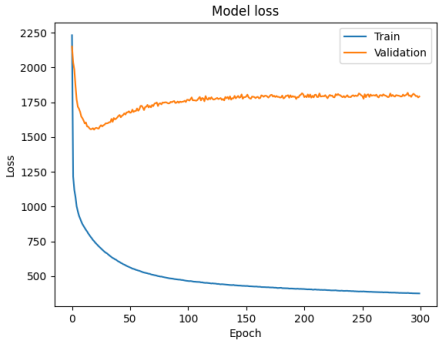


Figure B.7: 300 Epochs - VAE with LR = 0.0001, BS = 64, and LD = 256 units

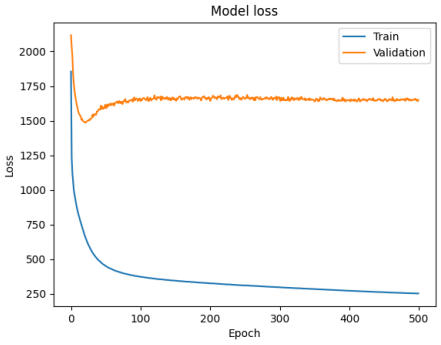


Figure B.8: 500 Epochs - VAE with LR = 0.0001, BS = 64, and LD = 256 units

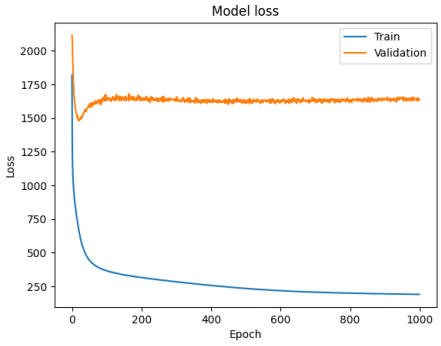


Figure B.9: 1000 Epochs - VAE with LR = 0.0001, BS = 64, and LD = 256 units

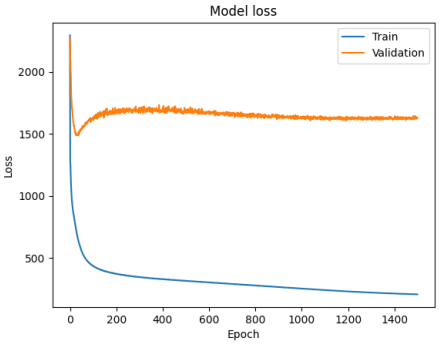
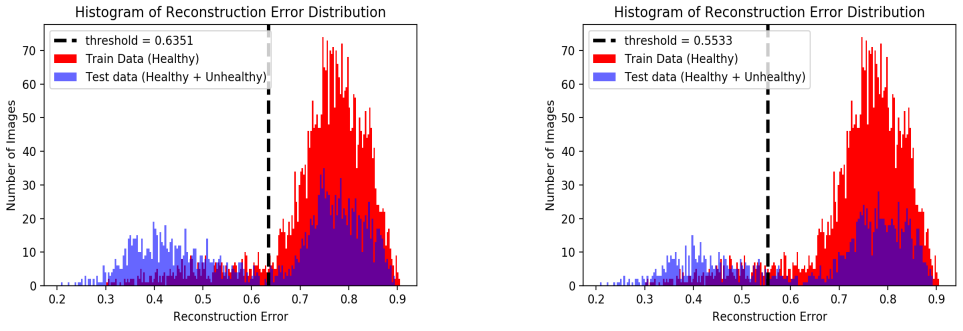


Figure B.10: 1500 Epochs - VAE with LR = 0.0001, BS = 128, and LD = 512 units

Appendix C

Reconstruction Error Histograms

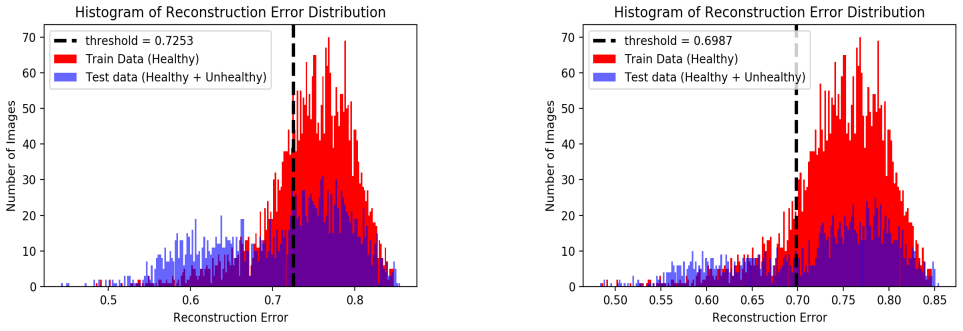
C.1 For VAE_{ssim}



(a) Balanced ADC data

(b) Imbalanced ADC data

Figure C.1: Reconstruction Error Histograms for ADC data with VAE_{ssim}

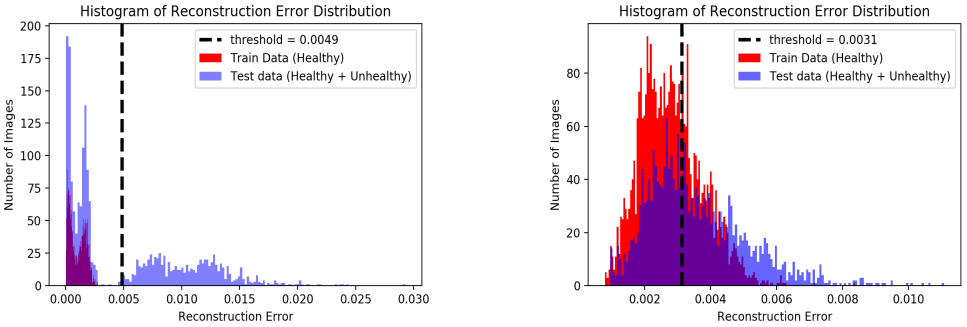


(a) Balanced T2w data

(b) Imbalanced T2w data

Figure C.2: Reconstruction Error Histograms for T2w data with VAE_{ssim}

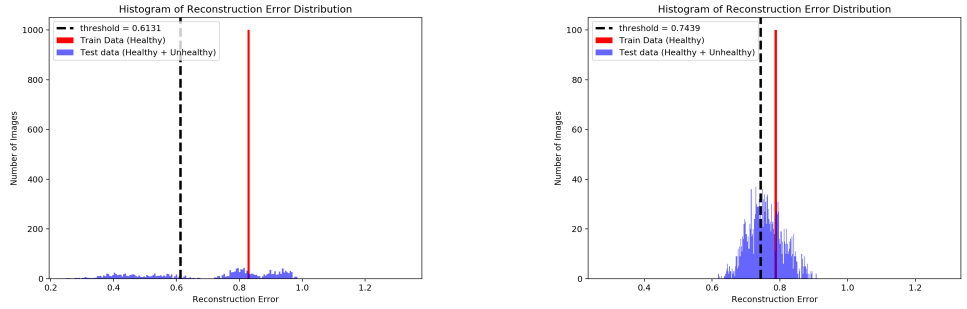
C.2 For AE_{mse}



(a) Balanced ADC data (b) Balanced T2w data

Figure C.3: Reconstruction Error Histograms for ADC and T2w data with AE_{mse}

C.3 For AE_{ssim}



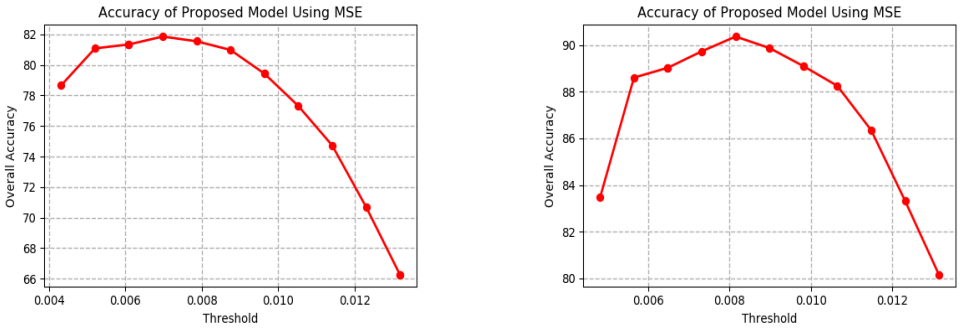
(a) Balanced ADC data (b) Balanced T2w data

Figure C.4: Reconstruction Error Histograms for ADC and T2w data with AE_{ssim}

Appendix D

Model Performance Metrics

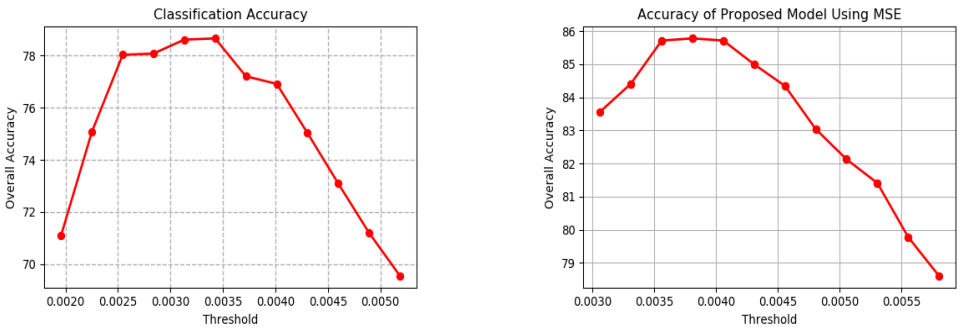
D.1 For VAE_{mse}



(a) Balanced ADC data

(b) Imbalanced ADC data

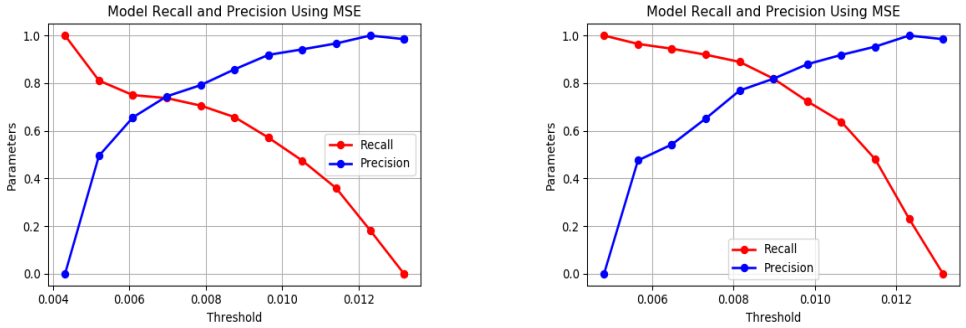
Figure D.1: Classification Accuracy for both configurations of ADC data with VAE_{mse}



(a) Balanced T2w data

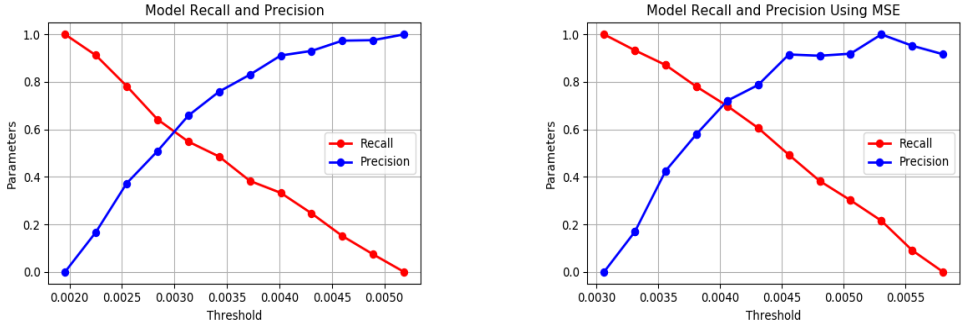
(b) Imbalanced T2w data

Figure D.2: Classification Accuracy for both configurations of T2w data with VAE_{mse}



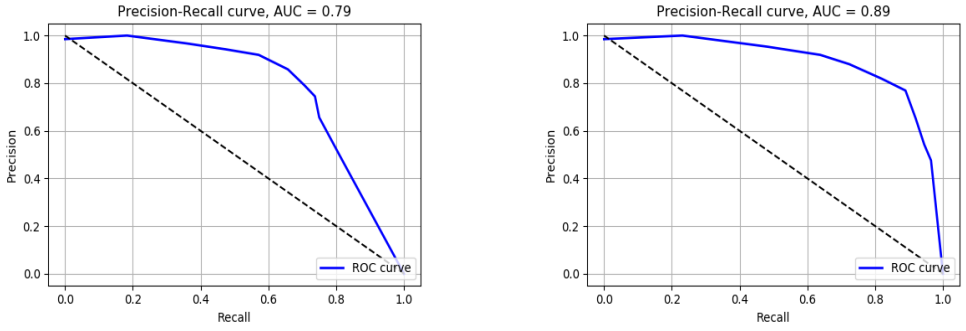
(a) Balanced ADC data (b) Imbalanced ADC data

Figure D.3: Recall and Precision for both configurations of ADC data with VAE_{mse}



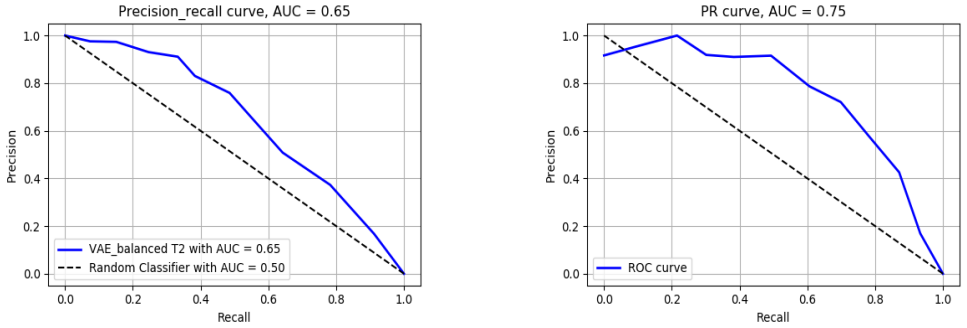
(a) Balanced T2w data (b) Imbalanced T2w data

Figure D.4: Recall and Precision for both configurations of T2w data with VAE_{mse}



(a) Balanced ADC data (b) Imbalanced ADC data

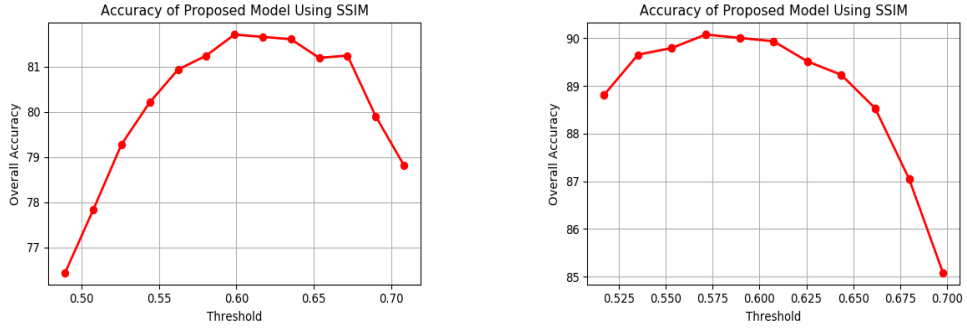
Figure D.5: Precision-Recall curve for both configurations of ADC data with VAE_{mse}



(a) Balanced T2w data (b) Imbalanced T2w data

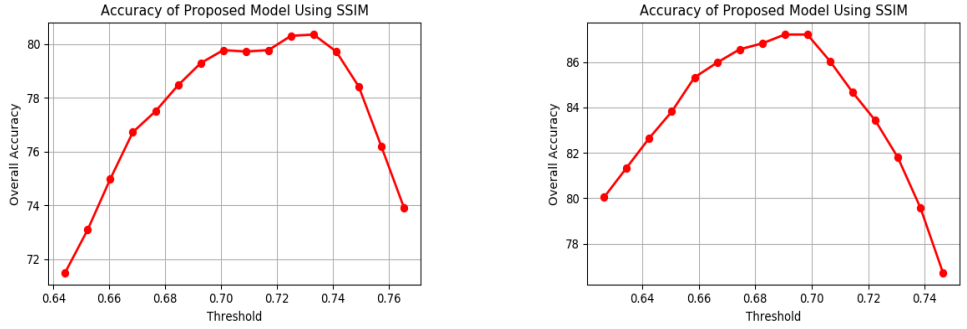
Figure D.6: Precision-Recall curve for both configurations of T2w data with VAE_{mse}

D.2 For VAE_{ssim}



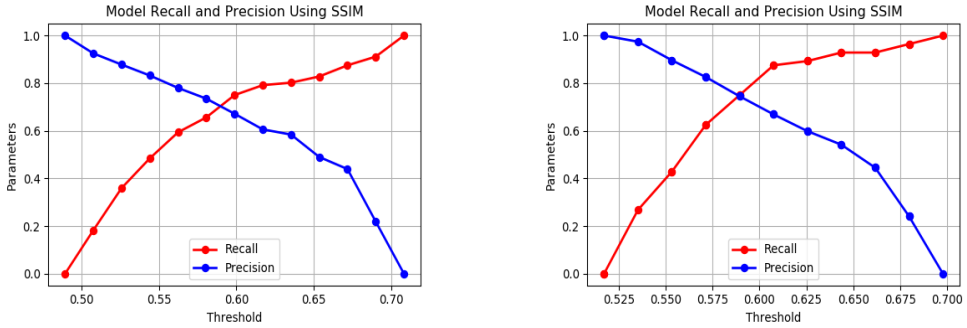
(a) Balanced ADC data (b) Imbalanced ADC data

Figure D.7: Classification Accuracy for both configurations of ADC data with VAE_{ssim}



(a) Balanced T2w data (b) Imbalanced T2w data

Figure D.8: Classification Accuracy for both configurations of T2w data with VAE_{ssim}



(a) Balanced ADC data

(b) Imbalanced ADC data

Figure D.9: Recall and Precision for both configurations of ADC data with VAE_{ssim}

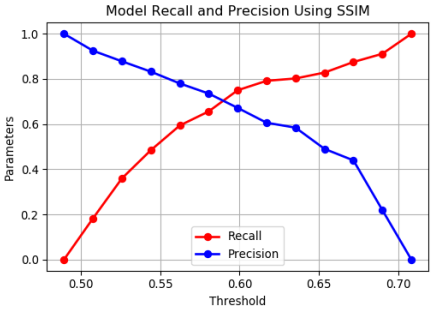


Figure D.10: Balanced ADC data

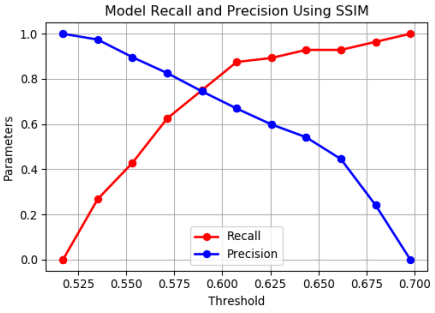
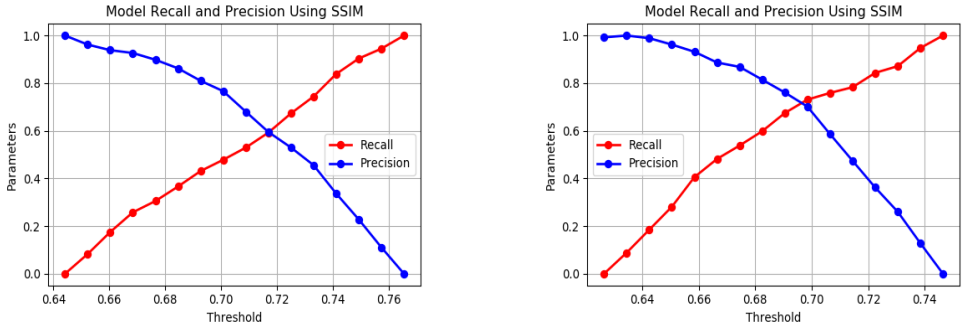


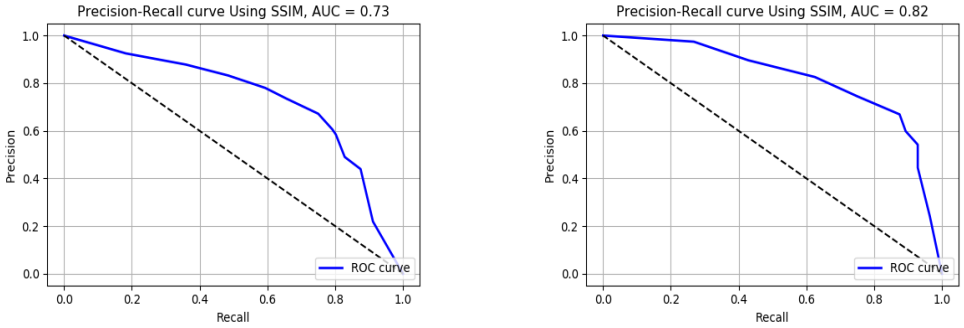
Figure D.11: Imbalanced ADC data



(a) Balanced T2w data

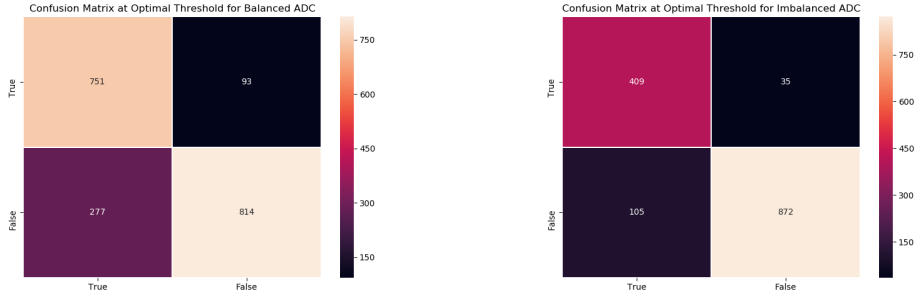
(b) Imbalanced T2w data

Figure D.12: Recall and Precision for both configurations of T2w data with VAE_{ssim}



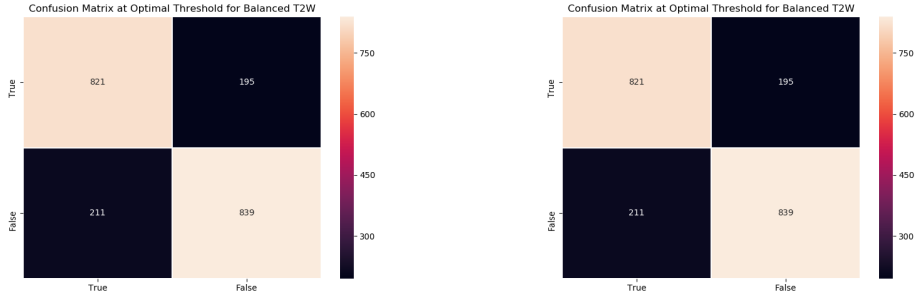
(a) Balanced ADC data (b) Imbalanced ADC data

Figure D.13: Precision-Recall curves for both configurations of ADC data with VAE_{ssim}



(a) Balanced ADC data (b) Imbalanced ADC data

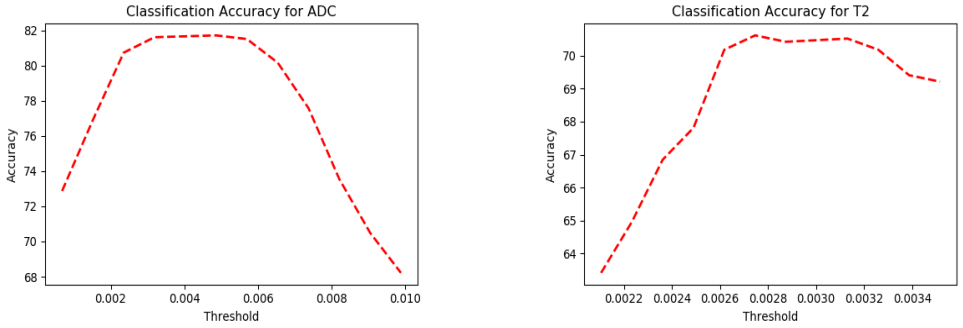
Figure D.14: Confusion matrix for both configurations of ADC data with VAE_{ssim}



(a) Balanced T2w data (b) Imbalanced T2w data

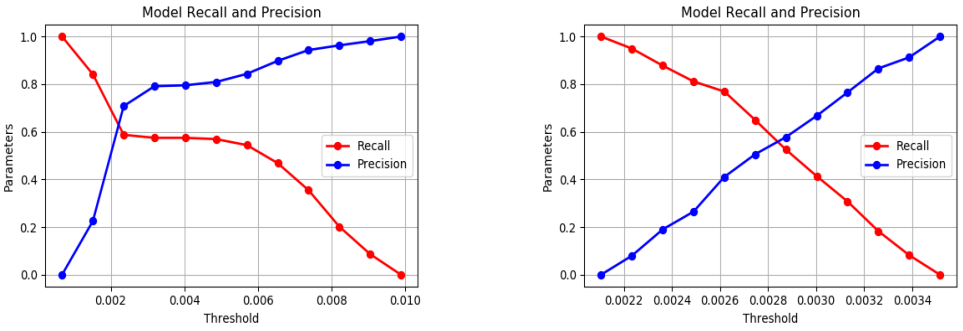
Figure D.15: Confusion matrix for both configurations of T2w data with VAE_{ssim}

D.3 For AE_{mse}



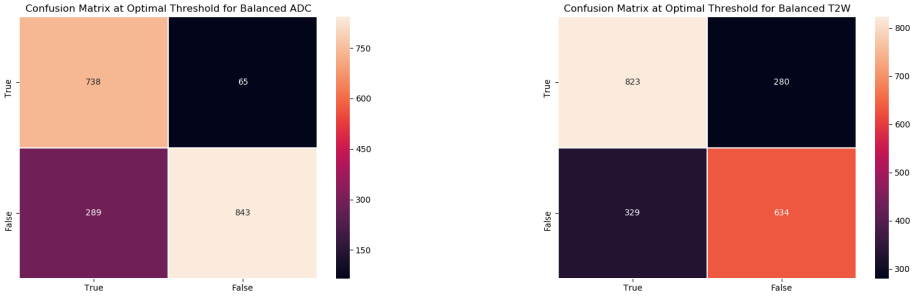
(a) Balanced ADC data (b) Balanced T2w data

Figure D.16: Classification Accuracy for balanced ADC and T2w data with AE_{mse}



(a) Balanced ADC data (b) Balanced T2w data

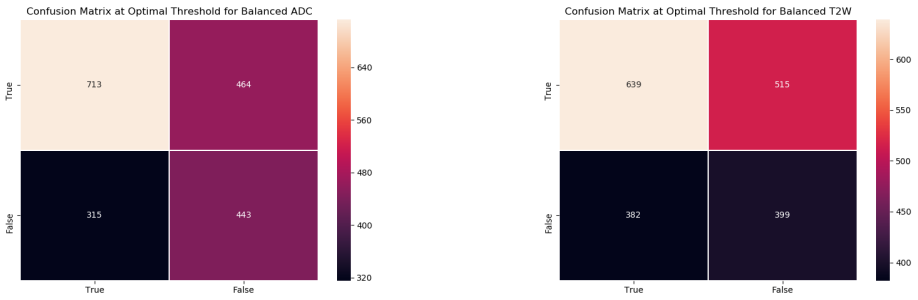
Figure D.17: Recall and Precision for balanced ADC and T2w data with AE_{mse}



(a) Balanced ADC data (b) Balanced T2w data

Figure D.18: Confusion matrix for balanced ADC and T2w data with AE_{mse}

D.4 For AE_{ssim}



(a) Balanced ADC data (b) Balanced T2w data

Figure D.19: Confusion Matrix for balanced ADC and T2w data with AE_{ssim}

Appendix E

Python Code

The relevant Python code for all task performed in the thesis is present on GitHub repository named "**Master_thesis**"and can be accessed using this link `https://github.com/haullah/Master_thesis.git` to the repository. The explanation of used libraries and python code is present in the **README.md** file in the repository.

Bibliography

[1] M. Ervik. F. Lam et al. J. Ferlay. Global cancer observatory: Cancer today.international agency for research on cancer, lyon 2018. URL `https://gco.iarc.fr/today`. [Online; accessed 15-May-2020].

[2] Jason A Efstathiou Freddie Bray MaryBeth B Culp, Isabelle Soerjomataram and Ahmedin Jemal. Recent global patterns in prostate cancer incidence and mortality rates. *European urology*, 77(1):38–52, 2020. ISSN 1873-7560.

[3] Bristol-Myers Squibb Oslo Economics. kostnader for pasientene, helsetjenesten og samfunnet. In Kreft i Norge, 2016.

[4] Marcelo Borghi Valeria Panebianco Lance A Mynderse Markku H Vaarala Alberto Briganti Lars Budäus Giles Hellawell Richard G Hindley et al.. Veeru Kasivisvanathan, Antti S Rannikko. Mri-targeted or standard biopsy for prostatecancer diagnosis. *New England Journal of Medicine*, 378(19):1767–1777, 2018.

[5] Abiodun Esther Omolara Kemi Victoria Dada Nachaat AbdElatif Mohamed Oludare Isaac Abiodun, Aman Jantan and Humaira Arshad. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), 2018. ISSN e00938.

[6] Pedro Domingos. A few useful things to know about machine learning. *Communications of the ACM*, 55(10:):78–87, 2018.

[7] Tahereh Hassanzadeh, Leonard G. C. Hamey, and Kevin Ho-Shon. Convolutional neural networks for prostate magnetic resonance image segmentation. *IEEE Access*, 7, 2019. doi: 10.1109/ACCESS.2019.2903284.

[8] B. Ravi Kiran, Dilip Mathew Thomas, and Ranjith Parakkal. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *CoRR*, abs/1801.03149, 2018. URL `http://arxiv.org/abs/1801.03149`.

[9] Stefan Bauer, Lutz Nolte, and Mauricio Reyes. Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization. *Medical image computing and computer-assisted intervention : MICCAI*, 14:354–361, 2011.

[10] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *CoRR*, abs/1906.02691, 2019. URL `http://arxiv.org/abs/1906.02691`.

[11] Xiaoran Chen and Ender Konukoglu. Unsupervised detection of lesions in brain MRI using constrained adversarial auto-encoders. *CoRR*, abs/1806.04972, 2018. URL `http://arxiv.org/abs/1806.04972`.

[12] Wouter Bulten and Geert Litjens. Unsupervised prostate cancer detection on h&e using convolutional adversarial autoencoders. *CoRR*, abs/1804.07098, 2018. URL `http://arxiv.org/abs/1804.07098`.

[13] Rong Yao, Chongdang Liu, Linxuan Zhang, and Peng Peng. Unsupervised anomaly detection using variational auto-encoder based feature extraction. pages 1–7, 2019. doi: 10.1109/ICPHM.2019.8819434.

[14] Jiahao Guo Lu Wang, Dongkai Zhang and Yuexing Han. Image anomaly detection using normal data only by latent space resampling. 10(23):1–19, 2020. doi: 10.3390/app10238660.

[15] Yoshua Bengio Ian Goodfellow and Aaron Courville. *Deep learning.* MIT press, 2016.

[16] Norsk Helseinformatikk. Prostatakreft., May 2019. URL `https://nhi.no/sykdommer/kreft/mannlige-kjonnsorganer-kreft/prostatakreft/`. [Online; accessed 1-May-2021].

[17] Cancer.Net Editorial Board. Digital rectal exam (dre), 2019. URL `https://www.cancer.net/navigating-cancer-care/diagnosing-cancer/tests-and-procedures/digital-rectal-exam-dre`.

[18] Steinar Valle Larsen. Exploring generative adversarial networks to improve prostate segmentation on mri. Master's thesis, University of Stavanger, Norway, June 2020.

[19] Wikipedia contributors. Gleason grading system, May 2020. URL `URLhttps://en.wikipedia.org/wiki/Gleason_grading_system`.

[20] Cancer research UK. Transrectal ultrasound scan (trus)., May 2019. URL `https://www.cancerresearchuk.org/about-cancer/prostate-cancer/getting-diagnosed/tests/transrectal-ultrasound-scan-trus`.

[21] Rais-Bahrami S. et al. Turkbey B Siddiqui MM. Comparison of mr/ultrasound fusion–guided biopsy with ultrasound-guided biopsy for the diagnosis of prostate cancer. *JAMA*, 313:390–397, 2015. ISSN 0098-7484.

[22] Choyke PL Cornud F Haider MA Macura KJ Margolis D Schnall MD Shtern F Tempany CM Thoeny HC Verma S. Weinreb JC, Barentsz JO. Pi-rads prostate imaging - reporting and data system: 2015, version 2. *European urology*, 69(1): 16–40, Jan 2016. ISSN 1873-7560. doi: 10.1016/j.eururo.2015.08.052.

[23] Martin J Graves Donald W McRobbie, Elizabeth A Moore and Martin R Prince. *MRI from Picture to Proton.* Cambridge university press, 2017.

[24] deSouza NM. Charles-Edwards EM. Diffusion-weighted magnetic resonance imaging and its application to cancer. *Cancer Imaging.*, 6(1):135–143, Sept 2006. doi: 10.1102/1470-7330.2006.0021.

[25] Robert Hecht-Nielsen. *Theory of the backpropagation neural network. In Neural networks for perception.* Elsevier, 1992.

[26] Trevor Darrell Ross Girshick, Jeff Donahue and Jitendra Malik. *Rich feature hierarchies for accurate object detection and semantic segmentation.* In Proceedings of the IEEE conference on computer vision and pattern recognition„ 2014.

[27] Ilya Sutskever Alex Krizhevsky and Geoffrey E Hinton. *Imagenet classification with deep convolutional neural networks.* In Advances in neural information processing systems., 2012.

[28] Aqeel Anwar. What is transposed convolutional layer?, March 2020. URL `https://towardsdatascience.com/what-is-transposed-convolutional-layer40e5e6e31c11`. [Online; accessed 30-May-2021].

[29] François Chollet et al. Keras., 2015. URL `https://keras.io/`. [Online; accessed 18-May-2021].

[30] François Chollet et al. Dense layer., 2015. URL `https://keras.io/api/layers/core_layers/dense/`. [Online; accessed 25-June-2021].

[31] Jason Brownlee. How to configure image data augmentation in keras., 2019. URL `https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/`. [Online; accessed 11-June-2021].

[32] Pierre Baldi. Autoencoders, unsupervised learning, and deep architectures. *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 27:37–49, 2012. URL `http://proceedings.mlr.press/v27/baldi12a.html`.

[33] Joseph Rocca. Understanding Variational Autoencoders (VAEs)., 2019. URL `https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73`. [Online; accessed 14-April-2021].

[34] Alessandro Beghi Marco Maggipinto, chiara Masiero and Gian Antonio Susto. A convolutional autoencoder approach for feature extraction in virtual metrology. *Procedia Manufacturing*, 17:126–133, 2018. URL `https://www.sciencedirect.com/science/article/pii/S2351978918311399`.

[35] Thushan Ganegedara. Intuitive Guide to Understanding KL Divergence., May 2018. URL `https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-understanding-kl-divergence-2b382ca2b2a8`. [Online; accessed 24-June-2021].

[36] Rodney LaLonde. Capsules for object segmentation (segcaps), file: metrics.py., 2020. URL `https://lars76.github.io/neural-networks/object-detection/losses-for-segmentation/`. [Online; accessed 18-June-2021].

[37] Gabriel Prieto et al. Renieblas. Structural similarity index family for image quality assessment in radiological images. *Journal of medical imaging.*, 4(3).

[38] Guido Van Rossum et al. Python programming language. *In USENIX annual technical conference.*, 41, 2007.

[39] Martín Abadi, Ashish Agarwal et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. [Online; accessed 02-June-2021].

[40] S Chris Colbert Stéfan van der Walt and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science Engineering*, 13:22–30, 2011.

[41] Ramswarup Kulhary. Opencv – overview, 2019. URL `https://www.geeksforgeeks.org/opencv-overview/`. [Online; accessed 19-June-2021].

[42] Jelle Barentsz Nico Karssemeijer Geert Litjens, Oscar Debats and Henkjan Huisman. Prostatex challenge data. *The Cancer Imaging Archive*, 2017. doi: 10.1109/TMI.2014.2303821.

[43] Smith K Freymann J Kirby J Koppel P Moore S Phillips S Maffitt D Pringle M Tarbox L Prior F. Clark K, Vendt B. The cancer imaging archive (tcia): Maintaining and operating a public information repository. *Journal of Digital Imaging*, 26(6), December 2013. doi: 10.1007/s10278-013-9622-7.

[44] Full Preprocessing Tutorial. Data science bowl, 2017. URL `https://www.kaggle.com/tzeny15/full-preprocessing-tutorial`. [Online; accessed 23-June-2021].

[45] Jason Brownlee. How to manually scale image pixel data for deep learning, 2019. URL `https://machinelearningmastery.com/how-to-manually-scale-image-pixel-data-for-deep-learning/`. [Online; accessed 03-March-2021].

[46] P.R. Jones. A note on detecting statistical outliers in psychophysical data. *Atten Percept Psychophys*, 81:1189–1196, 2019. doi: 10.3758/s13414-019-01726-3.

[47] Rhiannon van Loenhout, Frank Zijta, Robin Smithuis and Ivo Schoots. Prostate cancer - pi-rads v2, 2018. URL `https://radiologyassistant.nl/abdomen/prostate/prostate-cancer-pi-rads-v2`. [Online; accessed 29-June-2021].