

# CONVOLUTIONAL NEURAL NETWORKS FOR PROSTATE MAGNETIC RESONANCE IMAGE SEGMENTATION

By

Tahereh HassanZadeh Koohi

A THESIS SUBMITTED TO MACQUARIE UNIVERSITY

MASTER OF RESEARCH

DEPARTMENT OF COMPUTING

OCTOBER 2018



**MACQUARIE**  
University  
SYDNEY • AUSTRALIA

EXAMINER'S COPY

© Tahereh HassanZadeh Koochi, 2018.

Typeset in  $\text{\LaTeX}$  2<sub>ε</sub>.

# Statement of Originality

This work has not previously been submitted for a degree or diploma in any university. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

(Signed) \_\_\_\_\_

Date: \_\_\_\_\_

Tahereh HassanZadeh Koohi





# Dedication

To my **parents**,  
who dedicated all their life to me.



# Acknowledgements

First and foremost I would like to express my sincere appreciation to my supervisor, Dr Leonard Hamey for all his support, patience, friendly manner, motivation and immense knowledge. He has always been there to listen and give advice.

I am also incredibly grateful to my associate supervisor, Dr Kevin Ho-Shon, for all his immediate support, inspiration, encouragement and suggestions.

This research was undertaken with the assistance of resources and services from the National Computational Infrastructure (NCI), which is supported by the Australian Government.

Also, we would like to thank Richard Miller and Abed Kassis who have helped us to run all the jobs on NCI.



# Abstract

Digital medical image segmentation is the process of partitioning an image into several discrete and homogeneous regions. Segmentation is needed to find the boundary of the prostate either automatically or semi-automatically. One of the most accurate and non-invasive prostate imaging methods is Magnetic Resonance Imaging (MRI) which is usually employed for the prostate image segmentation and/or possible prostate anomalies detection.

In this research, to improve the Fully Convolutional Neural Network (FCNN) performance for prostate MRI segmentation, we analyse various structures of shortcut connections as well as the size of a deep network. We suggest six different deep 2D network structures for automatic MRI prostate segmentation based on FCNN. Our evaluations on the PROMISE12 dataset with ten-fold cross-validation indicate improved and competitive results. We analyse the results in detail, considering MRI slices, MRI volumes, test folds, and also the impact on prostate segmentation of using an EndoRectal Coil to capture the prostate MRI. Our best 2D network outperforms the state-of-the-art 3D FCNN-based methods for prostate MRI segmentation, without any further post-processing module nor pretraining on publicly available data.



# Contents

<b>Statement of Originality</b>	<b>iii</b>
<b>Dedication</b>	<b>v</b>
<b>Acknowledgements</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>List of Figures</b>	<b>xv</b>
<b>List of Tables</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Problem Description . . . . .	3
1.3 Objective of This Study . . . . .	3
1.4 Organisation of the Thesis . . . . .	4
<b>2 Background and Literature Review</b>	<b>7</b>
2.1 Atlas-Based Segmentation . . . . .	7
2.2 Shape-Based Segmentation . . . . .	9
2.3 Image-Based Segmentation . . . . .	10
2.4 Superpixel-Based Segmentation . . . . .	11
2.5 Deep Learning-Based Segmentation . . . . .	12

2.6	Discussion . . . . .	17
<b>3</b>	<b>Proposed Models</b>	<b>19</b>
3.1	Introduction . . . . .	19
3.2	Proposed Network Architecture . . . . .	20
3.3	Proposed Blocks Architectures . . . . .	23
3.3.1	Common Components . . . . .	25
3.3.2	Shortcut Connections . . . . .	27
3.4	Number of Features in the Proposed Models . . . . .	29
3.5	Conclusion . . . . .	32
<b>4</b>	<b>Data Analysis and Parameters Setting</b>	<b>33</b>
4.1	Dataset . . . . .	33
4.2	Data Normalisation . . . . .	34
4.3	Evaluation Metric . . . . .	36
4.4	Loss Function . . . . .	36
4.5	Hyper-parameter Setting . . . . .	37
4.5.1	Optimiser . . . . .	37
4.5.2	Dropout . . . . .	38
4.5.3	Size of the Network . . . . .	39
<b>5</b>	<b>Experimental Results</b>	<b>43</b>
5.1	Straight Model (Baseline) . . . . .	44
5.2	Bypass Model . . . . .	44
5.3	Output from All Model . . . . .	45
5.4	Input to All Model . . . . .	45
5.5	Dense Model . . . . .	46
5.6	Non-bypass Model . . . . .	46
5.7	Comparison of Normalisation Methods . . . . .	47
5.8	Analysing Unequal Layers per Block . . . . .	47
5.9	Quantitative Comparison . . . . .	48



---

5.10 Analysis of Data Folds . . . . .	51
5.11 Comparison with Prior Work . . . . .	52
5.12 Qualitative Comparison . . . . .	53
5.13 Analysing the EndoRectal Coil Effect . . . . .	53
<b>6 Conclusion and Future Work</b>	<b>57</b>
<b>A Appendix: Detailed Results</b>	<b>61</b>
<b>List of Symbols</b>	<b>73</b>
<b>References</b>	<b>77</b>



# List of Figures

2.1	The architecture of the U-Net and V-Net networks. . . . .	14
2.2	The architecture of ConvNet and BCNN networks. . . . .	15
3.1	Proposed network architecture for prostate segmentation. . . . .	21
3.2	Six proposed structures for the blocks. Con, Concatenation. . . . .	24
4.1	The training and validation error of ADAM optimiser in the Bypass model. . . . .	38
5.1	Comparison of the Output from All and Non-bypass models based on all MRI slices.	50
5.2	Comparison of the Output from All and Non-bypass models based on the size of the prostate. . . . .	50
5.3	Four sample images to show the quality of images. . . . .	51
5.4	The six sample segmented images using Non-bypass model. The red border is the ground truth and the green border in the predicted border. . . . .	54
5.5	Comparison of the Output from All and Non-bypass models based on obtain results per volume. Red bars, ERC-volumes; Blue bars, non-ERC volumes. . . . .	55
A.1	Comparison of loss error using different optimisation methods for the Bypass model.	63
A.2	Obtained results of the Output from All model on the all ten folds (test images). . .	65
A.3	Obtained results of the Non-bypass model on the all ten folds (test image). . . . .	66



# List of Tables

2.1	This table provide the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of the Atlas-based methods. . . . .	8
2.2	This table provide the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of the Shape-based methods. . . . .	9
2.3	This table provided the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of Image-Based methods. . . . .	10
2.4	This table provided the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of Superpixel-Based method. . . . .	11
2.5	This table provides the details of the Deep learning-based segmentation. . . . .	16
3.1	The connections pattern in the all six propose structures. I, Input of the block; O, Output of the block. . . . .	25
3.2	The quantity of feature maps for the first two blocks of the Straight, Bypass, Output from All, Input to All, Dense, and Non-bypass models. . . . .	31
4.1	This table provide the number of image slices for training, testing, and validation in the ten-fold cross-validation approach. Total, the total number of slices; Prostate, the number of slices contain prostate; ERC, the number of slices captured using EndoRectal Coil. . . . .	34
4.2	Comparison of original ERC and non-ERC images with their corresponding normalised images. NSep, Normalise sets Separately; NAll, Normalise All; NPix, Normalise Pixels. . . . .	35

4.3	Mean DSC of the Straight and Bypass models using the ADAM and SGD optimisers with different learning rates. ADAM1, learning rate is 0.01; ADAM2, learning rate is 0.001; ADAM3, learning rate is 0.0001; SGD1, learning rate is 0.01; SGD2, learning rate is 0.001; SGD3, learning rate is 0.0001. . . . .	38
4.4	Performance of using different locations and values for dropout based on mean DSC. Dropout1, drop out at the end of each layer with the probability of 0.2; Dropout2, Dropout at the end of each block with the probability of 0.2; Dropout3, dropout in the bottleneck with the probability of 0.5. . . . .	39
4.5	Comparison of the obtain results of using the different number of layers in the first three folds of all proposed models. . . . .	41
5.1	Performance of the Straight model using two layers (S2) three layers (S3) per block in all ten folds. M, Model; F, Fold. . . . .	44
5.2	Performance of the Bypass model using two layers (B2) three layers (B3) per block in all ten folds. M, Model; F, Fold. . . . .	44
5.3	Performance of the Output from All model by employing two layers (O2), three layers (O3), and also seven layers (O7) per block in all ten folds. M, Model; F, Fold. . . . .	45
5.4	Performance of the Input to All model using two layers (I2), three layers (I3), and five layers (I5) per block in all ten folds. M, Model, F, Fold. . . . .	46
5.5	Performance of the Dense model by employing two layers (D2), and three layers (D3) per block in all ten folds. M, Model; F, Fold. . . . .	47
5.6	Performance of the Non-bypass model by applying two layers (NB2), three layers (NB3), and seven layers (NB7) per block in all ten folds. M, Model; F, Fold. . . . .	47
5.7	Performance of the three normalisation methods on the Output from All model. NSep, Normalise sets Separately; NAll, Normalise All; NPix, Normalise Pixels M, Model; F, Fold. . . . .	48
5.8	Quantitative comparison of proposed models with another model. . . . .	48
5.9	Statistically significant comparison results using the Wilcoxon signed rank test. . . . .	49
5.10	The number of parameters of proposed networks. LPB, Layer Per Block; opt, Optimum number of layer. . . . .	50

- A.1 This table provide the Centers name, Filed strength, Resolution, and Imaging devices that used for collecting promise dataset. HUH, Haukeland University Hospital; BIDMC, Beth Israel Deaconess Medical Center; UCL, University College London; RUNMC, Radboud University Nijmegen Medical Center; ERC, EndoRectal Coil. . . . 62
- A.2 Comparison of using different values for the batch size based on mean DSC. Batch1, ADAM optimiser, learning rate 0.001, batch size 8; Batch2, ADAM optimiser, learning rate 0.001, batch size 16; Batch3, ADAM optimiser, learning rate 0.001, batch size 32; Batch4, ADAM optimiser, learning rate 0.001, batch size 64. . . . . 62
- A.3 Performance of using the different number of layers (two to nine) per block in the first three folds of all proposed models. . . . . 62
- A.4 Performance of using different number of blocks in the down-sampling and the up-samling parts in the all proposed models . . . . . 64
- A.5 Performance of the three normalisation methods on the Non-bypass model. NSep, Normalise sets Separately; NAll, Normalise All; NPix, Normalise Pixels M, Model; F, Fold. . . . . 64
- A.6 Performance of the Output from All and Non-bypass models with using various number of layers (4-4-5-7-10-12-15). M, Model; F, Fold . . . . . 64
- A.7 Segmentation of four different images from the test set of the first fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block. . . . . 67

- A.8 Segmentation of four different images from the test set of the second fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block. . . . . 68
- A.9 Segmentation of four different images from the test set of the fifth fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block. . . . . 69
- A.10 Segmentation of four different images from the test set of the sixth fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block. . . . . 70
- A.11 Segmentation of four different images from the test set of the tenth fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block. . . . . 71



# 1

## Introduction

Hospitals produce 50 petabytes of data annually [1]. According to International Business Machines (IBM) researchers estimation, medical images are at least 90 per cent of all medical data [2] making them the most prominent data in the healthcare enterprise. Dealing with the huge amount of medical images becomes overwhelming for radiologists, especially in some hospitals where they are faced with thousands of images daily. Therefore, automatic methods are needed to extract information from medical images. One of the most widely used methods for medical image analysis is image segmentation to find a specific organ or abnormality in the image. The purpose of this thesis research is developing an efficient method for automatic prostate MRI segmentation using deep neural networks.

### 1.1 Motivation

The prostate is a part of the male reproductive system that has an inverted pyramidal shape and usually weights between seven and sixteen grams. It measures about 3 cm in height and about 2.5 cm in diameter—approximately the size of an apricot [3]. The prostate can suffer from many diseases but most importantly prostate cancer. Prostate cancer is one of the more common causes

of death in developed countries [4]. According to Siegel et al. [5] there will be 164,690 new instances of prostate cancer and 29,430 deaths because of prostate cancer in the United States in 2018. Furthermore, the first cause of cancer and the second cause of cancer death in the United States of America is prostate cancer.

In Australia, according to the Prostate Cancer Foundation of Australia more than 3,000 men die because of prostate cancer annually—more than the number of women death due to the breast cancer [6]. The Australia Institute of Health and Welfare estimates that 17,729 new prostate cancer cases will be diagnosed in 2018, which would be 23.8% of all new male cancer cases in Australia. Also, they estimate that 3,500 men will die from prostate cancer, representing 12.7% of all cancer deaths in Australia [7, 8]. In 2017, prostate cancer was the third cause of cancer in Australia after Breast and Colorectal cancers and also the third cause of death after Lung and Colorectal cancer. [7, 8].

Early detection of prostate cancer can increase the chance of survival. The prevalence of prostate cancer elevates the importance of early-stage diagnostic and therapeutic methods. For example, in the United States, early-stage diagnosis of prostate cancer and improvements in therapeutic methods have decreased the rate of prostate cancer death by 40 to 50 per cent since the early 1990s [9].

One purpose of a diagnostic test is to detect the presence or absence of disease in an individual who may or may not have symptoms of disease. There are various methods that can be used to diagnose prostate cancer or other abnormalities in the prostate. MRI is a medical imaging technique that is the primarily tool for diagnosis and treatment planning for prostate ailments [10, 11]. The MRI device employs electric field gradients, strong magnetic fields, and radio waves to provide good contrast soft-tissue images. MRI images enable radiologists to obtain better lesion detection and staging for prostate cancer and MRI images allow for precise segmentation and accurate classification, and is a relatively harmless imaging method. [12]. Image segmentation is the first stage of analysis to find the prostate as well as possible prostate abnormalities the prostate MRI.

## 1.2 Problem Description

The process of segmenting an image into several discrete and homogeneous regions is image segmentation. It aims to alter the representation of an image to find a region or regions of interest in an image. Medical image segmentation is one of the most significant and active areas of research in medical image processing. The purpose of medical image segmentation is using a precise method to find the boundary of a specific organ or tissue, and it is a fundamental step for clinical studies including: diagnosis of disease, monitoring of organs or particular tissues, and, more importantly, treatment planning. Medical image segmentation is a difficult task because in most cases a specific organ has different shapes and sizes in different people [13]. Also, in some studies, the intensity value of the Region of Interest (ROI) is the same as the adjacent organs that can make segmentation even more challenging [14].

Medical image segmentation is usually done in one of three ways: manually, semi-automatically, or automatically. An expert radiologist can perform a manual segmentation of the ROI, but it is often time-consuming and tedious [15]. A further problem is that in some cases a radiologist may segment a specific image differently at various times or two radiologists may segment the same image dissimilarly [16]. However, when developing a semi-automatic or automatic segmentation method we almost always need ground truth images that should be created manually by expert radiologists. Even in the semi-automatic segmentation methods, an expert user is required to initialise or correct the segmentation. For example, the user can set a seed point or specify a region to start segmentation. In fully automatic segmentation, there is no human interaction during the segmentation of the image. In this type of segmentation, human knowledge is often employed to design an accurate method based on image processing and/or machine learning methods for image segmentation.

## 1.3 Objective of This Study

As discussed above, the most accurate and safe method for recognition of any types of abnormalities in the prostate gland is employing MRI image. Finding the boundary of the prostate in the MRI image is fundamental to recognise possible disease in the prostate. Because of the considerable diversity in size, appearance, shape, and texture of the prostate and the lack of a clear prostate

boundary, especially in malignant prostate tissues, prostate segmentation is a challenging problem even for expert radiologists. Regarding the increasing incidence of prostate cancer in the world and the importance of prostate cancer detection in the early stage for patients survival, finding an effective method for prostate segmentation and eventually, prostate cancer detection is necessary.

One of the methods for analysing of medical images is Deep Learning [17] that is a part of machine learning methods [18]. A deep neural network is constructed from multiple layers of neurons for feature extraction and classification. Improvement in machine learning methods especially deep learning convinced researchers to use deep learning in computer vision applications and specifically in medical image analysis such as image segmentation [19, 20].

Fully Convolutional Neural Network (FCNN) is a type of Convolutional Neural Network (CNN) that has been introduced for image segmentation [21]. The purpose of the FCNN is to create an output image analogous to the ground truth of the input image. U-net [22] and DenseNet [23] are two of FCNN-based networks for medical and natural image segmentation respectively. In our research, we try to develop new segmentation methods based on U-net and DenseNet structures.

In this thesis, we suggest six different structures with a particular focus on using various pattern of shortcut connections [24] as well as varying the size of the networks for automatic 2D MRI prostate image segmentation. We evaluate the performance of the following six network structures: Straight, Bypass, Output from All, Input to All, Dense, and Non-bypass models. After extensive experiments, and analysing the results in detail, our best model (Non-bypass) is found to outperforms the state-of-the-art 3D FCNN-based prostate segmentation methods. We show that, using shortcut connections can also decrease the accuracy of the network; therefore, it is critical to use shortcut connections in the proper place in the network. Therefore, starting and ending points of the shortcut connections also critical. In addition, we find that the quality of the training images has a significant effect on the final results.

## 1.4 Organisation of the Thesis

The remainder of the thesis is organised as follows. In chapter 2, different MRI prostate image segmentation methods are categorised into five main groups and discussed. In chapter 3, we first analyse the key issues in developing new networks for segmentation. We then introduce the six FCNN-based networks for MRI prostate image segmentation. In chapter 4, we explain the

dataset and data normalisation methods that will be employed for segmentation, and investigate the hyper-parameter settings for network training. In [chapter 5](#), the outcomes of the six suggested models for MRI prostate segmentation are discussed, and the best models are identified. Finally, in [chapter 6](#), we summarise the key outcomes and discuss propose future work.



# 2

## Background and Literature Review

Established semi-automated and automated prostate image segmentation models can mainly be categorised into four various groups: Atlas-based, Shape-based, Image-based, and Superpixel-based segmentation. Recently, Deep learning-based approaches have achieved state-of-the-art results in image processing and specifically in image segmentation [25–28]. In this chapter, some significant papers that analyse 2D MRI images or 3D MRI volumes for prostate image segmentation will be presented. At the end of each section, we provide tables and a discussion section to analyse the performance of the segmentation models. The Dice Similarity Coefficient (DSC) [29–32] was chosen for performance comparison because it is a common metric for evaluation of image segmentation results. Some papers combine the mentioned methods; we discuss these papers in one section based on the main method that they used.

### 2.1 Atlas-Based Segmentation

One of the methods that can segment unseen images by using manually labelled training images is Atlas-based segmentation [33]. In this approach a registration method is used to align the atlas image or images to a new image. There are two different atlas-based methods, namely

Parametric and Non-parametric methods. Parametric atlas methods typically incorporate the manually segmented training images into a single atlas image [34], whereas Non-parametric models apply all of the training images individually to create multiple atlas images [35].

The Demons registration algorithm can be used to create a so-called probabilistic atlas [36]. The Demons registration method computes the pixel value transformation between a reference image and several moving images. To do this, the registration employs two stages; that, intensity-based affine transformation, and then non-rigid demons registration. For example, Ghose et al. [37], and Gao et al. [38] utilise a probabilistic atlas and the random forest algorithm for prostate image segmentation. They use several distinct random forest classifiers to identify the prostate boundary. Finally, leveraging the probabilistic representation of each pixel, the multi-image graph cuts algorithm is used to obtain the final segmentation. Also, Ghose et al. [39] suggested a hybrid method including a probabilistic atlas model and Statistical Shape and Appearance Model (SSAM) for 3D prostate image segmentation. Li et al. [40] proposed another method based on the probabilistic atlas and an enhanced random walk algorithm for prostate segmentation. Another segmentation method based on probabilistic atlas introduced by Martin et al. [41]. In the first stage, images are registered to the probabilistic atlas; in the second step, the information from the first step is merged to obtain a deformable surface defining the prostate boundary. Finally, a supervised atlas-based segmentation method has been proposed using the combination of adaptive Active Appearance Model (AAM) for coarse segmentation followed by a Support Vector Machine (SVM) for fine segmentation of prostate images [42].

In the above models, different algorithms are used for finding the location of the prostate automatically. However, some researchers manually identify the appropriate prostate

region and apply their algorithms only on the ROI. Korsager et al. [43] manually extracts the prostate as a rectangular ROI and uses the ROI and its corresponding label to create an atlas. This work uses, both shape

Ref	DSC%	Year	Dataset Size	Segmentation	Data
[37]	91	2012	15 volumes	Automatic	3D
[39]	89	2012	15 volumes	Automatic	3D
[38]	88.98	2014	107 images	Automatic	2D
[43]	88	2015	67 images	Automatic	3D
[42]	87.5	2014	40 volumes	Automatic	2D
[41]	84	2010	36 volumes	Automatic	3D
[40]	80.7	2013	30 volumes	Automatic	3D

TABLE 2.1: This table provide the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of the Atlas-based methods.



information which is extracted from atlas registration, and intensity information that is derived from histograms of the image. Finally, the graph cut is used for globally minimising the energy function that is extracted from the Maximum A Posteriori (MAP) segmentation.

The summary of the discussed Atlas-based methods can be seen in Table 2.1. In this table, the reference number, performance (mean DSC), publication year and the size of the datasets are listed. Comparison of these papers is difficult because they employed different datasets for evaluation. However, based on the obtained results, using atlas with statistical shape and appearance method [37] is the best result, with 0.91 average mean DSC. Also, some papers use the combination of Atlas methods and deep learning for prostate image segmentation that will be explained in the deep learning section.

## 2.2 Shape-Based Segmentation

In this method, a template shape based on control points along the boundary of the ROI is needed. Active Shape Model (ASM) [44] and Active Appearance Model (AAM) [45] are two common shape-based techniques. These methods use derived landmarks, which can be specified either manually or automatically, for segmentation. A combination of MAP and AAM are used for segmentation in several papers [46–48]. These papers use MAP for estimation of a new log-likelihood function and different types of descriptors to find the prostate boundary. For example, Firjani et al. [47] apply a visual appearance descriptor, a 3D spatially rotation-variant descriptor (the output of descriptor will change with rotation of the image), and a homogeneity descriptor to find the boundary of the prostate.

Lastly, a 3D shape descriptor is applied for separating the prostate from the background. Also, they propose a later version of their work, replacing the rotation-variant descriptor with a rotation-invariant descriptor [48].

Ref	DSC%	Year	Dataset Size	Segmentation	Data
[47]	92	2011	270 volumes	Automatic	3D
[49]	88	2011	108 volumes	Semi-Automatic	3D
[48]	85.5	2011	180 volumes	Automatic	3D
[50]	81.79	2013	40 volumes	Automatic	3D
[46]	80	2011	28 images	Automatic	3D

TABLE 2.2: This table provide the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of the Shape-based methods.

The Active Appearance Model is another shape-based model that is employed in various papers for prostate and/or its components (peripheral and central zones) segmentation [49, 50].

As is shown in Table 2.2, Firjani et al. [47] obtain 0.92 average mean DSC as the best result. Recently, researchers employ the combination of these methods with an atlas-based method as discussed in the previous section, or with deep learning that will be discussed below for segmentation.

## 2.3 Image-Based Segmentation

Some methods begin their segmentation with an initial template then refine it based on the image data while minimising error. The Active Contour Model (ACM) or Snake and its derivatives are image-based [51]. Snake is an energy-minimising framework that can find edges, lines, and boundaries and can be utilised in shape recognition, object tracking, segmentation, and edge detection. Skalski et al. [52] suggested a novel application of ACMs with gradient vector flow for segmentation. They use typical prostate shape as prior knowledge to improve the accuracy of their model. Liew et al. [53] proposed another segmentation method based on 3D MRI images. They present a novel rotational volume slicing method along with a contour shrinking technique.

Some researchers apply the combination of contour-based algorithm and classifiers for prostate delineation. For instance, Yang et al. [54] introduced a novel method based

References	DSC%	Year	Dataset Size	Segmentation	Data
[54]	91.45	2016	72 images	Automatic	2D
[55]	83	2017	22 volumes	Semi-automatic	2D
[56]	78.4	2017	PROMISE12	Automatic	2D
[52]	-	2013	8 images	Automatic	3D
[53]	-	2015	10 volumes	Automatic	3D
[57]	-	2013	33 volumes	semi-automatic	2D

TABLE 2.3: This table provided the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of Image-Based methods.

on a modification of a level set formulation. Firstly, they segment a medium slice of the prostate image manually to provide prior information, then by using similarity estimation they detect the prostate. Lastly, the contour of the prostate is acquired by the enhanced level set model. In this approach, for every new image, the location of the prostate is specified with a registration schema.

In the iterative model, the algorithm starts from a specified pixel or region, then iteratively

refines the boundary with or without radiologists or the guidance of the other experts. Wu et al. [57] suggested a two-stage algorithm for 3D MRI prostate segmentation. First, an initial surface mesh of the prostate is acquired interactively then the graph cut algorithm is used to find the prostate. Another method uses a learning approach for localisation of the prostate by learning global context [56]. In this paper, fine segmentation is performed by min-cut on the sparse spherical graph of the prostate. In other work, Wang et al. [55] designed a two-stage model for prostate image partitioning. The first stage is the calculation of the multi-view label-relevance probability map. The next phase of their method includes collaborative clustering that is including the calculation of a membership function, entropy function, weight calculation, and cluster centre calculation to learn to segment the pixels of the image into background and foreground groups. In the image-based methods, the mean DSC is not available for some of the methods. The summary of the results can be seen in Table 2.3.

## 2.4 Superpixel-Based Segmentation

Xiaofeng Ren and Jitendra Malik introduced the concept of superpixel in 2003 [58]. In this method, instead of using individual pixels for image segmentation, groups of pixels with similar colours or grey levels are considered. One of the precise methods for superpixel segmentation is Simple Linear Iterative Clustering method (SLIC) [59]. In this method, the k-means clustering algorithm [60] is used for clustering pixels. Superpixel methods have been used for prostate segmentation, but the prostate is only a small part of the image and using grey level based features cannot correctly find the prostate. Therefore, the superpixel method should be used with some other techniques to detect the superpixels that represent the prostate.

For example,  
in Mahapatra  
et al. [63]  
and Mahapatra  
et al. [62] they  
apply the SLIC

References	DSC%	Year	Dataset Size	Segmentation	Data
[61] (Level set)	89.3	2016	PROMISE12	Semi-automatic	3D
[62]	81	2014	PROMISE12	Automatic	2D
[63]	80	2013	PROMISE12	Automatic	2D

TABLE 2.4: This table provided the references, performance, publication year, the size of the dataset, Type of Segmentation, and Type of Data of Superpixel-Based method.

<sup>1</sup>PROMISE12 is a published prostate MRI segmentation dataset with 50 volumes and their corresponding labels.

method, and then create an adjacency graph of the superpixels. To merge the adjacent superpixels, they employ a graph cut minimisation method, and finally, a 3D level set method is used to find the boundary of the prostate. Also, Tian et al. [61] uses a random 3D active contour model along with superpixel method for segmentation.

All three methods are applied to the PROMISE12 dataset for prostate segmentation. Mahapatra et al. [63] obtain 0.91 DSC, outperforming the others [61, 62] who achieve 0.81 and 0.87 per cent mean DSC respectively. The detailed information can be seen in Table 2.4

## 2.5 Deep Learning–Based Segmentation

Improvement in machine learning methods especially deep learning has convinced researchers to use deep learning in computer vision applications [19, 20]. A deep neural network is constructed from multiple layers of neurons such that each layer learns to transform its input data into a new more abstract. In particular, a Convolutional Neural Network (CNN) [64] is a kind of deep network that has been successfully applied for visual image processing. The leading operator of CNN is convolution, consisting of learnable filters or kernels that are convolved across the input image, computing the dot products between the filters and the receptive fields to produce feature maps [65]. In this section, we review papers that apply the combination of the CNN and other methods for MRI segmentation.

A successful hybrid method for prostate image segmentation uses an atlas as well as deep learning. Cheng et al. [66] proposed a hybrid method combining an atlas–based active appearance method along with a deep learning method to improve 3D MRI prostate image segmentation. They apply AAM for estimating the prostate boundary then use deep CNN to refine the boundary. In the first phase, they separate the atlas into various groups based on a similarity measure. Each image slice is assigned to the most similar atlas group and they then employ AAM training in each subgroup to find a boundary around the prostate. In the second phase, they extract 2D  $64 \times 64$  image patches around the AAM predicted boundary. They use pre–trained AlexNet [67] to classify the patches into prostate and non–prostate to refine the boundary.

Cheng et al. [68] presented another work using both patch–based and holistic (image to image) deep learning methods for prostate image segmentation. In this paper, they employ a Holistically Nested Network (HNN) architecture for image–based segmentation. For training

their network they first crop 25% of images from top, bottom, left, and right to find the prostate area. Then the Coherence Enhanced Diffusion (CED) filter is used to enhance the quality of the prostate boundaries. In the end, both the original MRI images and the CED–MRI images with their corresponding labels are used for training the HNN for the prostate image segmentation.

Recently Jia et al. [69] proposed a course–to–fine segmentation method using an atlas method and deep learning. In this paper, a registration–based segmentation is used to find the approximate boundary of the prostate; then they extract image patches around the prostate region to find the prostate boundary by applying deep network VGG–19 [70] and LeNet–5 [71]. In the paper, they fine–tune pre–trained VGG–19 for finding the prostate boundary. Also, to show the efficiency of utilising pre–trained networks they train LeNet–5 from scratch using the extracted image patches. In the end, they conclude that using the pre–trained network is more precise than their separately trained network for prostate image segmentation. In related work, He et al. [72] proposed a three–level coarse–to–fine segmentation method. In the first level, the 3D volume of interest is extracted by employing 3D Haar features then an Adaptive Feature Learning Probability Boosting Tree (AFL–PBT) voxel classifier is used to classify pixels into three groups: near, interior, and exterior. Finally, CNN is used to refine the prostate boundary.

Proposal–based segmentation is another well known model for natural image segmentation that has been applied for prostate image segmentation [73]. In this approach, an image is divided into several patches or proposals then the proposals that contain prostate are separated. For example, Yan et al. [73], first generate a set of prostate proposals by using the Geodesic Object Proposal (GOP) algorithm [74] for 3D segmentation of the prostates then a graph is used to select highly effective proposals. Finally, CNN is employed to detect highly effective features to refine the boundaries. Two other types of networks that apply for prostate image segmentation are Stacked Sparse Auto Encoder (SSAE) [75] and Independent Subspace Analysis (ISA) networks [76]. The SSAE uses a sparse patch matching method, and the ISA employs sparse label propagation method for feature extraction for prostate image segmentation.

Almost all of the papers discussed above use a combination of various image processing and machine learning methods for feature extraction, coarse segmentation and fine segmentation. However, recently some researchers employ only CNN for both feature detection and segmentation. Fully Convolutional Neural Network (FCNN) is a version of CNN that is designed for image

segmentation [21]. FCNN is constructed from two parts including down-sampling (encoding, convolution) and up-sampling (decoding, deconvolution). In some networks, there is a specific block named Bottleneck (Bridge) to connect these two parts. In the down-sampling section, the network tries to extract features as it goes from the higher resolution to lower resolution while the up-sampling part attempts to reconstruct the coarse-to-fine segmentation with transposed convolution [77]. FCNN utilises an end-to-end method for learning. For 2D images, it uses image-to-image, and for 3D volumes, it applies volume-to-volume supervised learning.

One of the first articles that uses FCNN with 2D convolution for 2D medical semantic image segmentation is U-Net [22]. As seen from Figure 2.1a, U-Net is constructed from three parts. The down-sampling part contains four blocks such that between each pair of blocks, there is a max-pooling layer to select the maximum value of the cluster and to halve the size of the feature maps

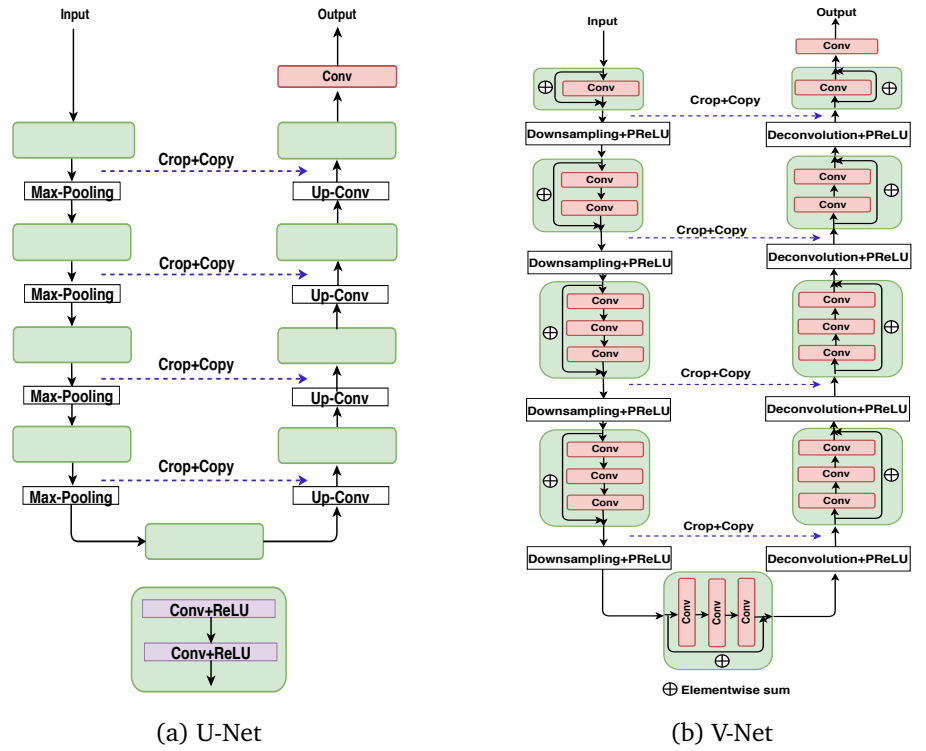


FIGURE 2.1: The architecture of the U-Net and V-Net networks.

[78]. In the up-sampling section, between each pair of blocks there is a convolution layer with  $2 \times 2$  kernel size to double the size of the output feature maps. Additionally, the Bottleneck block connects the two parts. All the blocks are constructed from two convolution layers followed by non-linearity and finally, a  $1 \times 1$  convolution layer is used in the last layer. Also, to improve the results in this structure, long connections are used for cropping and copying a part of the extracted feature maps from the down-sampling part and concatenating them with the obtained feature maps from the up-sampling section. Recently, A new version of U-Net for prostate MRI image segmentation was introduced [79]. In this paper, they try to improve the accuracy of the

network by amplifying the length of the U–Net by adding a  $1 \times 1$  convolution layer in each block. Additionally, this paper uses dropout to overcome overfitting.

In 2016, the first version of 3D FCNN for segmentation of 3D volumes of prostate images was presented as V–Net [80] (see Figure 2.1b). In this network, there are four blocks in each part similar to U–Net, but the number of layers in each block is different. In each block, there is a residual shortcut connection for summing the input feature maps and the output feature maps of the block, element by element. It means that by applying element–wise sum the number of feature maps will be constant and the result of summing two groups of feature maps will be sent to the next stage. Also, four long connections are used to concatenate the feature maps from the first part to the second part, as in U–Net. In this network, instead of max–pooling, they employ convolution with the kernel size of  $2 \times 2$ .

Another paper for 3D MRI prostate segmentation is ConvNet [81] (see Figure 2.2a). The aim of this paper is to analyse the effect of using short and long residual connections. In this network, each residual block is constructed from two convolution layers with kernel size of  $3 \times 3$  and they use element–wise sum to combine the input

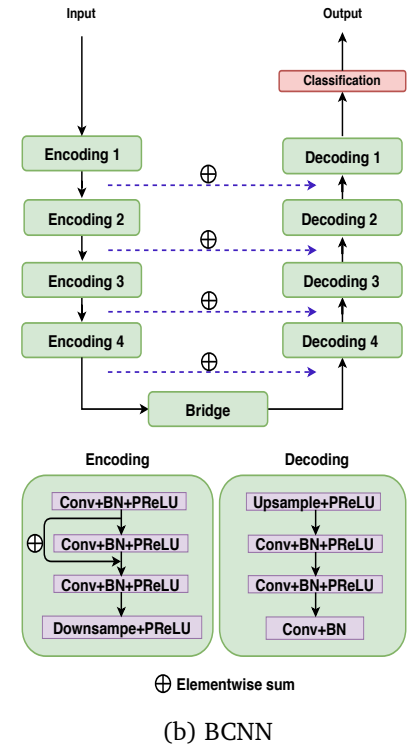
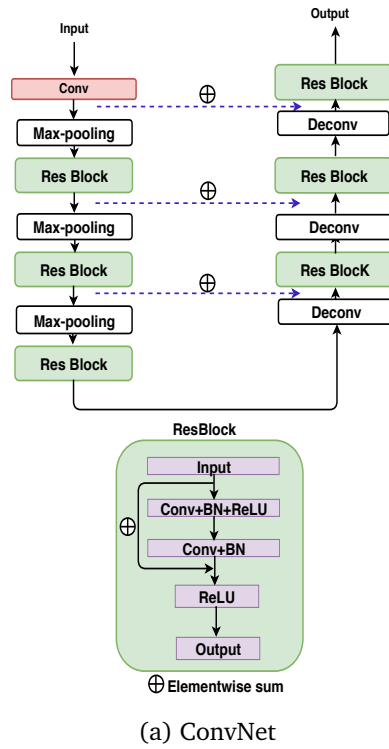


FIGURE 2.2: The architecture of ConvNet and BCNN networks.

put of the second convolution layer as the short residual connection before applying the non–linearity. Moreover, they utilise long connections to sum the extracted feature maps from the first part of the network to the second part. In this paper, they train the network as variants with only short or long and with both residual connections. They show that using the combination of short and long residual connections is more effective for prostate image segmentation.



Mun et al. [82] proposed a new network for 3D MRI prostate segmentation call the Baseline Convolutional Neural Network (BCNN) (see Figure 2.2b). In this paper, different structures are employed in the encoding and decoding parts. All the blocks contain three layers of convolution, but in the encoding blocks, there is a shortcut connection to sum the output of the first layer with the output of the second layer. A corresponding connection does not exist in the decoding part. In the encoding section, downsample implies a convolution layer with stride two and upsample in the decoding block indicates the deconvolution operator. Also, to reuse the extracted feature maps of the first part in the second part, they utilise long connections and element-wise sum. The primary purpose of the paper is the comparison of six different types of objective functions: the Jaccard Index, Hamming Distance, Euclidean Distance, Cosine Similarity, Dice Coefficient, and Cross Entropy. Using the same architecture, the results show that Cosine Similarity is the best and the Dice Coefficient is the second best among the six objective functions to train the network for prostate image segmentation.

In summary, some papers used the combination of the CNN with other methods such as atlas registration. However, recent work has employed only FCNN for coarse to fine prostate image segmentation and achieved positive results. With regards to the dataset that they used, and the publication years, we

Ref	DSC	Year	Dataset Size	Segmentation	Data
[66]	92.5	2016	120 volumes	Automatic	3D
[69]	91	2018	PROMISE12 PROSTATEX17 <sup>2</sup> [83]	Automatic	2D
[68]	89.77	2017	250 volumes	Automatic	2D
[73]	89	2016	PROMISE12	Automatic	2D
[79]	88.5	2017	1324 images	Automatic	2D
[75]	87.4	2016	66 images	Automatic	3D
[80]	86.9	2016	PROMISE12	Automatic	3D
[81]	86.9	2017	PROMISE12	Automatic	3D
[76]	86.7	2013	30 images	Automatic	2D
[82]	85.37	2017	PROMISE12	Automatic	3D
[72]	84	2017	PROMISE12	Automatic	3D

TABLE 2.5: This table provides the details of the Deep learning-based segmentation.

can understand that the obtained results outperformed most of the previous methods that used

<sup>2</sup>PROSTATEX17 is a published prostate MRI classification dataset with 204 volumes with Gleason scores representing prostate cancer diagnosis but without segmentation ground truth [83]. All studies included T2-weighted (T2W), proton density-weighted (PD-W), dynamic contrast enhanced (DCE), and diffusion-weighted (DW) imaging. The images were acquired on two different types of Siemens 3T MR scanners, the MAGNETOM Trio and Skyra.



the combination of some other algorithms for the prostate image segmentation (see Figure 2.5). Cheng et al. [66] achieved the best result with 92.5 mean DSC in this section.

## 2.6 Discussion

There are many papers with various methods for semi-automatic and automatic MRI prostate segmentation. We have only discussed some of them in this section. For comparison of the papers, a critical factor is the dataset that they used for segmentation. Except for the PROMISE12 and PROSTATEX17 datasets, all the other datasets are unpublished.

To evaluate an MRI prostate dataset, we should consider three important points: total number of volumes or images, number of slices that contain the prostate, and number of EndoRectal Coil (ERC) and non-EndoRectal Coil (non-ERC) images. In all of the unpublished datasets, the number of volumes or images are provided. However, there is not any information about the number of slices with and without prostate. Using a dataset with more diverse prostate image scans or using images with a clearer prostate region can increase the accuracy of the proposed model. Another important factor is using ERC. This device is placed into the rectum to obtain high-quality images during 1.5T MRI imaging, however it create spikes and bright regions in the MRI that can decrease the accuracy of the prostate image segmentation. In the unpublished datasets, there is not any information about the number of the ERC or non-ERC images, yet this is a significant factor in the accuracy of the prostate image segmentation.

In the absence of information, we suppose that the number of the prostate and non-prostate images and also the number of ERC and non-ERC images in all of the datasets are reasonably consistent. In our review, six papers obtained mean DSC greater than 90% of which four applied their methods on unpublished datasets with 15, 270, 18, and 120 volumes [37, 47, 54, 66]. Among those, Cheng et al. [66] obtained the best results by using atlas-based registration and deep CNN methods with 92.5 mean DSC. Also, Jia et al. [69] obtained mean DSC of 91% by using PROMISE12 datasets with atlas-based registration and deep CNN; however, they applied extra post-processing for boundary refinement. Mahapatra et al. [63] achieved equivalent results by using the superpixel method.



# 3

## Proposed Models

In this chapter, the aim is to design a precise model for 2D MRI prostate segmentation by using Fully Convolutional Neural Network (FCNN). To do this, we analyse the different parameters of the CNN, including convolution, Batch Normalisation (BN), dropout, its size, and more importantly the role of using the shortcut connections in the prostate image segmentation. Afterwards, six network structures are proposed for automatic 2D MRI prostate image segmentation. To the best of our knowledge, there is no similar extensive work to analyse FCNN for MRI prostate segmentation in the literature.

### 3.1 Introduction

There are critical issues to design a deep neural network such as the size of the network and the components (convolution, max-pooling, etc.) that are typically employed to develop a new model. First, we present these challenges, and then we will explain our proposed models to address these problems for MRI prostate segmentation.

One of the straightforward methods to enhance the precision of the FCNN may be to increase the depth of the network. In FCNN, the depth depends on the number of blocks in the network

and the number of layers in each block [84].

When expanding the size of the network, two important issues have to be considered. The first one is the feasibility of the implementation of the network because it can expand the memory usage and the computational time. The second vital issue is overfitting [85]. Increasing the depth of the network can enlarge the number of parameters<sup>1</sup>, and the model will be more prone to overfitting. Overfitted (also called overparameterized) networks have more parameters than can be fitted by the data. To address this problem, the network needs to be trained with more data. Unfortunately, one of the main problems in medical image processing is a shortage of labelled data.

Besides the size of the network, the architecture of the network is important to obtain an accurate segmentation. CNNs are constructed from different components including convolution, Batch Normalisation (BN), and dropout. Another component of the network may be the shortcut and/or long connections that can have a considerable effect on the results. The design of the network including the place of its components and their parameters will significantly affect the network's performance.

Some papers have analysed different parameters consisting of the depth of the network, and the effect of using  $1 \times 1$  convolution on the network for image classification [84] and image recognition [70], also the role of short and long residual connections has been investigated for bio-medical image segmentation [81, 86], but until now, no article has been published to investigate the role of the size of the network and other parameters such as different structures of the shortcut connections for image segmentation.

### 3.2 Proposed Network Architecture

To address the issues listed in the previous section, we propose models based on U-Net [22] to define an accurate FCNN-based model for prostate image segmentation. U-Net is a relatively straightforward network that has been introduced for 2D medical image segmentation. Our models are based on U-Net because they are constructed from three parts like U-Net, consisting of down-sampling, bottleneck, and up-sampling. Similar to U-Net, we consider the MRI images

---

<sup>1</sup>Number of parameters for each convolution layer is  $(N \times M \times X + 1) \times Y$ . Where  $N \times M$  is the size of the filter;  $X$  is the number of input feature maps;  $Y$  is the number of output feature maps. The count 1 represents the bias term.

as 2D slices and use 2D convolutions. However, the architecture of our proposed models and their components are different from U-Net.

We suggest a relatively deep FCNN network structure. The diagram of this architecture is shown in Figure 3.1. As is shown, this network is constructed from three parts including down-sampling, bottleneck, and up-sampling. Six blocks can be seen in the down-sampling part and six blocks in the up-sampling section. In the bottleneck, there is another block to connect the two parts. Each block has several components that will be discussed in Section 3.3.

As shown in Figure 3.1, in the down-sampling part, the input image is directed into the first block and the output feature maps (convolution output [65]) of the first block are fed to the next block as the input. This process is repeated several times. After each block in the down-sampling part, a max-pooling operator [78] halves the size of the feature maps. In the bottleneck, there is a block that connects the down-sampling section to the up-sampling part. In the up-sampling section, each block is preceded by a deconvolution layer (also called transposed convolution) [77] to double the dimension of the feature maps. In our proposed models, we use stride along with padding for transposed convolution. It means, perform zero padding on the input feature maps and then apply convolution to increase the resolution (dimension) of the feature maps.

In Figure 3.1, the resolution of the feature maps (the first two numbers) along with the output rate (the number of feature maps output by each layer (the third number)) is specified. As is shown, in the down-sampling part the resolution decreases after each max-pooling operator to extract information and in the up-sampling section, the resolution will be increased to the original size of the image. The last layer is a  $1 \times 1$  convolution

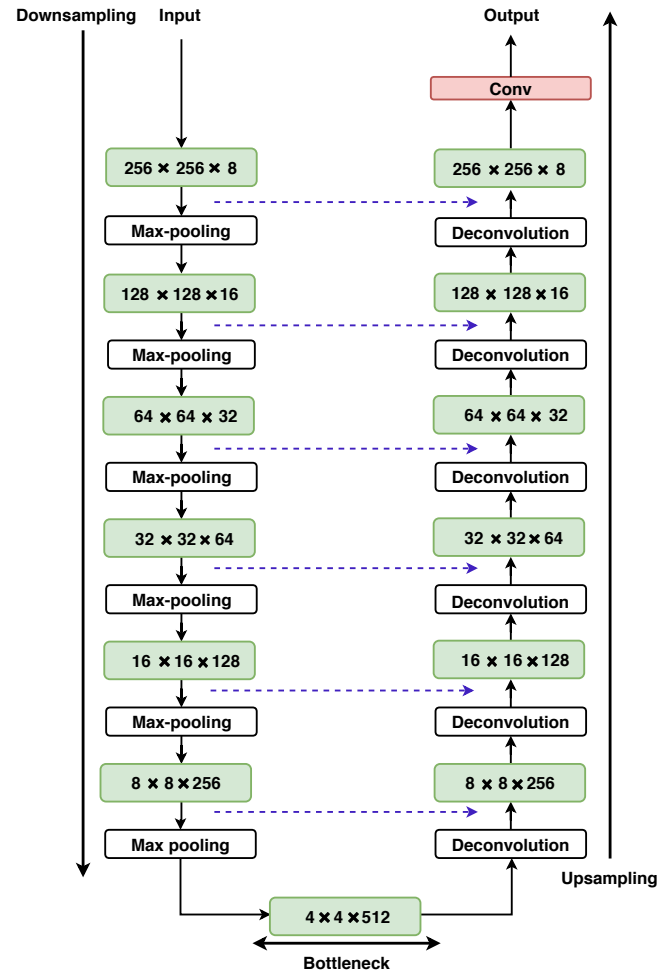


FIGURE 3.1: Proposed network architecture for prostate segmentation.

layer with one output channel that produces the output segmentation image. The dimension of the input image and the output segmentation image is the same ( $256 \times 256$ ).

One of the problems of using deep networks is degradation (increasing the depth of the network can decrease the accuracy). Because of using multiple convolution layers and max-pooling in the down-sampling process a part of the spatial information is lost [24, 81]. Therefore, feature maps in the up-sampling part will have more information deficiency. To improve the quality of feature maps in the up-sampling section the extracted data from the down-sampling part can be reused by using *long* connections or *highway* connections [87]. Srivastava et al. [87], introduced the highway network to create the deep network by using highway connections. Highway or long connections let feature maps flow across several layers. In our proposed architecture we use a long connection between each block in the down-sampling part and its corresponding block in the up-sampling section to bring high spatial resolution information across to be combined with deconvolved lower resolution information in the up-sampling part of the network. Our proposed architecture uses six long connections (shown with dash lines see Figure 3.1) to copy extracted feature maps to the up-sampling section. The first Long connection is the longest, connecting the output feature maps of the first block to the last block, and the sixth one is the shortest.

As seen in Figure 3.1, we propose a deep FCNN with several layers that could cause problems by increasing the number of parameters and run time of the network. To limit these problems, we specify the number which we call the output rate. This is the third number on each block in Figure 3.1. The output rate is the number of output feature maps of each convolution layer in the block. As shown in Figure 3.1, we set the output rate of the first block as eight; it means that every layer in the first block has eight output feature maps regardless of the number of input feature maps, and then the output rate doubles in the following blocks in the down-sampling part. The reason for starting the output rate with a small number is the size of the network—using a higher output rate can increase the number of parameters. With regards to the limited available training data, increasing the number of parameters without considering the network components and the number of training images can increase the possibility of overfitting.

In the down-sampling part, the deeper blocks have a higher output rate to represent more complex features. We employ max-pooling to halve the spatial dimensions of the feature maps after each block [22]. Max-pooling does not change the number of feature maps.

The Bottleneck is the most profound block in our network with the highest output rate (512) and capacity for the most complex features. It connects the down-sampling part to the up-sampling part and has the smallest feature map size ( $4 \times 4$ ). Structurally, the bottleneck is otherwise the same as the other blocks.

In the up-sampling part after applying each deconvolution operator, the size of the feature maps doubles. While each block halves the number of feature maps based on the output rate.

### 3.3 Proposed Blocks Architectures

Another issue that can be very significant in the final segmentation results is the architecture of the blocks. We propose six different structures for the blocks. As shown in Figure 3.2, all six block models are constructed from three layers and each layer includes a convolution layer with the kernel size of  $3 \times 3$  follow by a Rectified Linear Unit (ReLU) activation function [88]. To improve the generalisation of the network, batch normalisation [89] and dropout [90] are also employed. However, as will be explained in chapter 4, we apply dropout in different locations with various values along with changing the number of layers to study the effects on the segmentation.

As shown in Figure 3.2, we apply various structures of the shortcut connections [24] to investigate their effects on the prostate image segmentation. To understand how different layers are connected, we provide Table 3.1 to show the connectivity patterns of our proposed models. Each sub-table has four columns representing the sources of a possible connection—the Input of the block (I) and the outputs of the three layers. Each of the four rows represents a possible connection destination—one of three layers within the block and the final output of the block (O). The \* shows where there is a direct or intermediary (also called chain) connection between two corresponding layers.

For instance, in Table 3.1a, the input image is directed to the first layer, the output of the first layer into the second layer, the output of the second layer into the third layer, and finally, the output of the third layer is assigned to the output of the block. The output of the block is pointed to the next step which is most often Max-pooling or Deconvolution. As shown in Figure 5.4a and Table 3.1a, there are not any shortcut connections in Model 1. We named it straight because the feature maps flow through the layers in the block one after another without using any

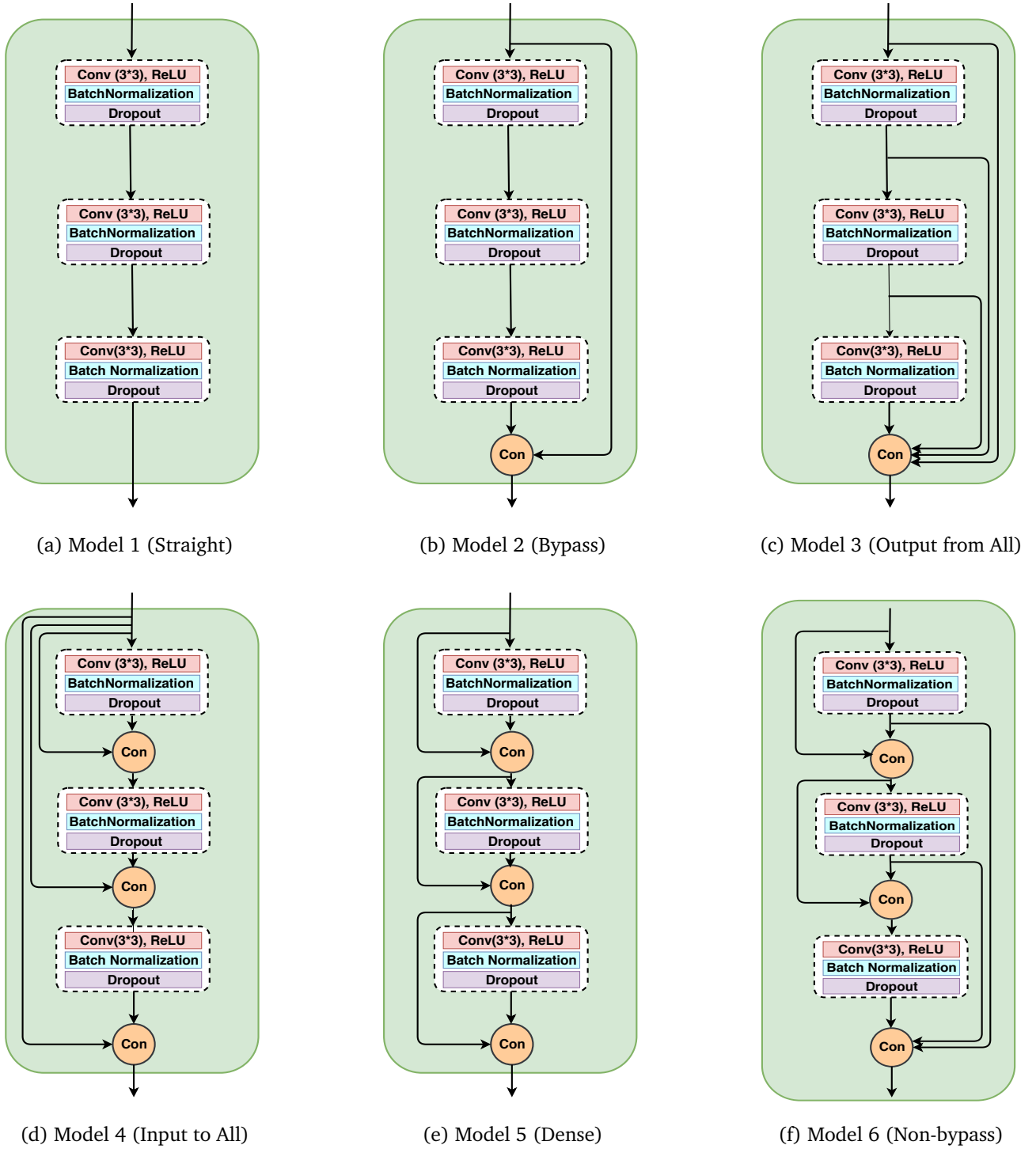


FIGURE 3.2: Six proposed structures for the blocks. Con, Concatenation.

skip connections. The Straight model is the baseline architecture that we use to understand the effect of using shortcut connections in the other five models. In those models, we employ different patterns of shortcut connections among the layers. To create shortcut connections we employ the concatenation operator to collect feature maps and send them to the specific location in the



From	I	1	2	3
To				
1	*			
2		*		
3			*	
O				*

(a) Model 1 (Straight-Baseline)

From	I	1	2	3
To				
1	*			
2		*		
3			*	
O	*			*

(b) Model 2 (Bypass)

From	I	1	2	3
To				
1	*			
2		*		
3			*	
O	*	*	*	*

(c) Model 3 (Output from All)

From	I	1	2	3
To				
1	*			
2	*	*		
3	*		*	
O	*			*

(d) Model 4 (Input to All)

From	I	1	2	3
To				
1	*			
2	*	*		
3	*	*	*	
O	*	*	*	*

(e) Model 5 (Dense)

From	I	1	2	3
To				
1	*			
2	*	*		
3	*	*	*	
O	*	*	*	*

(f) Model 6 (Non-bypass)

TABLE 3.1: The connections pattern in the all six propose structures. I, Input of the block; O, Output of the block.

network. For example, in Table 3.1b, besides the baseline connections in the Straight model there is a shortcut connection between the input of the block and the output of the block to transfer the input feature maps of the current block to the next block. This is called the Bypass model because it allows the block to be bypassed.

In the following, we first explain the common operators that we utilise in each layer of the blocks including dropout, batch normalisation, activation function, and the kernel size. We then describe each model separately based on the shortcut connections structure.

### 3.3.1 Common Components

Since we have a deep network and our training images are limited, we should employ some approaches to control overfitting including dropout [90] and batch normalisation [89] to create sparsity in our proposed models that can accelerate the training and improve the accuracy of the network.

Dropout can be an efficient method to exclude the complication of co-adaptations on the training data and also functions as a regulariser. Dropout randomly deletes a portion of the features by omitting hidden layers units with a specified probability [90]. We analyse the role of the dropout in our proposed architectures by using it in the different locations with various

probabilities. As is shown in Figure 3.2, in all of the proposed structures, we use dropout at the end of each convolution layer as a baseline configuration. In chapter 4, we will apply dropout only at the end of each block (one dropout per block), and finally, just in the bottleneck (one dropout after each layer in the bottleneck block).

Another feature in our proposed models is Batch Normalisation (BN) [89] for data normalisation during the training of the network. Data normalisation is one of the most critical parts of training a network. It is common to normalise input data before training the network, but after applying the convolution operator and non-linearity, the distribution of the data will be changed. The purpose of BN is normalising the output of network layers during training, and it is known that this normalisation can accelerate the training of the network [89]. According to Ioffe et al. [89], to calculate BN, each mini-batch should be normalised to zero mean and unit variance. BN starts with zero mean and unit variance normalisation, but during the training, can learn other parameters that might be better for normalisation. The BN algorithm is presented in Appendix A, an Algorithm A.1.

One of the reasons for using BN is reducing the covariance shift [89] i.e. the changing distribution of the test data versus the training data [91]. If a network is trained with X as the input images and Y as the corresponding labels with a particular distribution, the network could learn the distribution of the training samples. If the network tested with new images from a different distribution, the results can be very poor. For example, training the network with black cats images and testing the network with white or coloured cat images would likely have poor performance. similarly, our network may not generalise well enough to recognise new unseen samples from a different distribution [92]. Moreover, internal covariance shift can happen during the backpropagation [89]. If the parameters of the first layer of the network change, it could be the change the distribution of the second layer, and consequently, it changes the layer outputs as well [89]. Deep networks, such as our proposed models are more prone to the problem.

It is also known that BN can work as a regulariser similar to dropout [89]. Therefore, if BN is used in the network, dropout value selection should be made more carefully otherwise more information will be lost [93]. However, dropout cannot be replaced with BN because the effect of the BN on overfitting is less than dropout; therefore, a proper solution is using both of them simultaneously as we are doing in our proposed models.

As the activation function, our proposed models employ Rectified Linear Unit (ReLU) [88] for all layers within the blocks. ReLU was first used for training of deep networks in 2011 [94]. Activation function defines the output of each unit with regards to its input, and the ReLU defines the positive part of its argument ( $f(x) = \max(0, x)$ ). ReLU can decrease the probability of the vanishing gradient [95] during the back propagation [94].

In the last layer of our proposed network, we apply the Sigmoid function [96] as the non-linearity. The sigmoid commonly employed for two-class classification. The equation of this function can be seen in the Equation 3.1.

$$f(x_i) = \frac{1}{1 + \exp^{-x_i}} \quad (3.1)$$

The convolution layers of our networks employ small  $3 \times 3$  convolutional filters in the convolution layers. Using a stack of very small  $3 \times 3$  receptive fields is more efficient than using bigger receptive fields like  $5 \times 5$  or  $7 \times 7$  [70]. Utilising the small kernel the network will apply more non-linear layers, and it can decrease the number of model parameters [70].

In summary, in the proposed networks we use  $3 \times 3$  kernel for convolution layers, apply ReLU as the activation function in the hidden layers after each convolution layer, and utilise BN after activation function for all six models. Dropout is also used to control overfitting. In the next section, we discuss the key issue of shortcut connections.

### 3.3.2 Shortcut Connections

To improve the spatial information of the feature maps, in addition to using long connections, the input feature maps of each block and the feature maps that are created within the block can be reused applying shortcut connections [24]. He et al. [24], connected each pair of layers by using element-wise sum as the shortcut connections. The output of the element-wise sum of the two feature maps will be a feature map with the same dimension as the input feature maps. A shortcut connection allows the error signal to be backpropagated to another layer in the block directly that can be helpful to address the vanishing gradient phenomenon in the network. Besides, it can make the training process of the network easier and faster, and also it can pass the extracted details among the layers inside of the block and/or from the beginning of the block to the end of it [97].

In our work, we use concatenation to create the shortcut connection as well as long connections

in all of our proposed networks. Concatenation stacks the feature maps on top of each other. Let  $U_{ij}$  be the output feature maps of the specified layer and  $V_{ij}$  is the skipped feature maps that come from a particular section of the network by using shortcut connections, the concatenation of the mentioned features can be shown as Equation 3.2.

$$x_{i,j} = \begin{pmatrix} U_{ij} \\ V_{ij} \end{pmatrix} \quad (3.2)$$

To study the effect of using concatenating shortcut connections for prostate MRI segmentation, we propose six different structures of connections. At first, as is shown in Figure 5.4a, we suggest a simple model called Straight as the baseline without using any shortcut connections. In this model, there are three layers per each block such that the output of each layer is fed the next layer as the input. This simple block can show the learning capability of a network without any shortcut connections. Further, by comparison of the Straight model results with the other models, it will be possible to understand whether a network with shortcut connections can outperform networks without shortcut connections.

In the Bypass model (see Figure 5.4b), we employ a concatenation operator at the end of the block to collect the input feature maps of the block at the end of the block (*bypass* connection). This connection directing the provides input of the current block to the next block.

In the Output from All model, as seen in Figure 5.4c, we add a concatenation operator at the end of the block to collect four sets of feature maps including the outputs of the all layers in the block (*gathering* connections) and the input of the block (*bypass* connection) and feed them to the next block. Using this model, we can explore whether using feature maps with different levels of information increases the learning ability of the network.

In the Input to All model shown in Figure 5.4d, the blocks input feature maps are provided to each of the layers (*scatter* connections). We use three concatenations per block, such that each of them concatenates the output feature maps of the previous layer and the input of the block and sends them to the next layer. The final output is also concatenated with the input. In other words, this model uses both *bypass* and *scatter* connections. By comparing the results of the Bypass model with the Input to All model, we can understand whether reusing the input feature maps, within the block can improve the results. Also, by comparing the results of the Input to All model and the Output from All model, it will be possible to understand whether it is better to reuse the block's

input via scatter concatenations or to gather the outputs of the layers.

Our Dense model (see Figure 5.4e) follows the the Dense model structure [23]. The Dense model was introduced by Huang et al. [23] for natural image classification and obtained promising results. In the Dense model, after each layer, there is a concatenation operator that can concatenate the output of all previous layers and the original input of the block. The fully convolutional Dense model was applied for colour image segmentation by Jegou et al. [98] and obtained positive results for natural image segmentation.

In our work, we use the fully dense block for MRI prostate segmentation, with block structure is shown in Figure 5.4e. In the original dense network, there are direct connections between layers, but we use chain connections to deliver the same effect while decreasing the number of shortcut connections. For example, to transfer the input feature maps of the block to the third layer, the feature maps are first moved to the second block by the first shortcut connection and the first concatenation operator. Then, the second shortcut connection transfers both the outputs of the first layer and the input feature maps of the block to the second concatenation.

The Dense model uses four types of connections: bypass, scatter, gathering and *internal*. Internal connections are those which start and end within the block. A dense block as defined by Huang et al. [23], is fully connected within the block, but when it is made into a Densenet a shortcut is added around the block. Thus, our proposed Dense model with a bypass connection is exactly equivalent.

Lastly, the Non-bypass model (see Figure 5.4f) is equivalent to the dense model except that, there is no connection between the input and the output of the block. This means that the input feature maps of the block will not send to the next block. Each layer receives the output of all previous layers and the input of the block using the chain connections. The output feature maps of all layers are concatenated to each other at the end of the block and then sent to the next block. This model uses gathering, scatter, and internal connections. Comparison of the results of the Dense and Non-bypass models will show the effect of the bypass connection.

### 3.4 Number of Features in the Proposed Models

To understand the effect of the concatenation and output rate on the number of feature maps we provide Table 3.2. The table shows the number of features maps as input, the output of each

layer, the output of each concatenation, and also the output of each block for the first two blocks of all six proposed network structures.

As is shown in Table 3.2, in all of the structures the input image enters the network and then, based on the structure of the network, different numbers of features are created. In the Straight model the number of output feature maps of each layer, and also the output of the block is the same as the output rate of that block because there are not any shortcut connections to increase the number of feature maps.

In the Bypass model, the number of output feature maps for each block is the sum of the number of input feature maps of the block and the number of output feature maps of the last layer of the block. However, the quantity of output feature maps of each layer in each block is equal to the output rate. For instance, in the Bypass model, the number of input feature maps of the second block is nine, and the output of the last layer is 16; as a result, 25 feature maps will be sent to the next block.

In the Output from All model, all the previous feature maps both those created within the block and those that entered the block, are collected and sent to the next block, therefore, 25 feature maps are collected at the end of the first block and 73 at the end of the second block.

In the Input to All model, after each layer, there is a concatenation operator to collect the features of the previous layer and the input of the block. The output feature maps of the layers are the same as the output rate, but with increasing the depth of the network, feature reuse is increased. For example, in the second block, nine feature maps entered the network and were reused three times in the block.

In the Dense model, the number of feature maps passing within the block is more than other models, and it shows that with using the same amount of layers this structure will create the highest number of feature maps among our proposed structures. The difference in the Non-bypass model is that there is no connection between the input and output of the block. For this reason, the number of feature maps decreases to 48 at the end of the second block, whereas the Dense model has 73 feature maps at the end of that. With the increasing depth of the network, the differences between the Dense and Non-bypass models will be considerable.

As shown in Table 3.2, the Straight model has the lowest number of feature maps, but with the added shortcut connections, the number of feature maps increases in the other five proposed

Block number	Layer Name	Straight	Bypass	Output from All	Input to All	Dense	Non-bypass
Block 1	Input of the block	1	1	1	1	1	1
	Output of Layer 1	8	8	8	8	8	8
	Output of Con 1	-	-	-	9	9	9
	Output of Layer 2	8	8	8	8	8	8
	Output of Con 2	-	-	-	9	17	17
	Output of Layer 3	8	8	8	8	8	8
	Output of the Block	8	9	25	9	25	24
Block 2	Input of the block	8	9	25	9	25	24
	Output of Layer 1	16	16	16	16	16	16
	Output of Con 1	-	-	-	25	41	40
	Output of Layer 2	16	16	16	16	16	16
	Output of Con 2	-	-	-	25	57	56
	Output of Layer 3	16	16	16	16	16	16
	Output of the Block	16	25	73	25	73	48

TABLE 3.2: The quantity of feature maps for the first two blocks of the Straight, Bypass, Output from All, Input to All, Dense, and Non-bypass models.

models. The Input to All, Dense, and Non-bypass models generate more feature maps within the blocks compared to the three other models because of employing concatenation operators between the layers. Furthermore, in the Bypass and Input to All models, the quantity of feature maps that transfer between the blocks is the same, but the number of feature maps within the block is different because unlike the Bypass model that employs one concatenation, the Input to All model using three concatenations within the blocks. Additionally, the Output from All and Dense models have the same number of output feature maps at the end of the blocks, but internally they produce and employ different numbers of feature maps. By comparing, the results of the Bypass versus Input to All, and also the Output from All against Dense model we can understand whether creating and employing more feature maps within the blocks can improve the results.

Finally, the Straight, Bypass and Output from All models have the same structure within the block and produce the same number of features inside the blocks, but the type and number of feature maps that they transfer between the blocks are different. In this case, by comparing the discussed methods, we can understand how the type and number of feature maps that move between the blocks affect the final segmentation results.

### 3.5 Conclusion

We have proposed six FCNN-based networks for prostate MRI segmentation with a particular focus on using different patterns of shortcut connections consisting of: the Straight structure without any shortcut connections (baseline model), Bypass, Output from All, Input to All, Dense, and Non-bypass networks. We have compared the suggested structures and considered the type and number of feature maps that they create and use during the convolution process in the first two blocks.



# 4

## Data Analysis and Parameters Setting

In this section, firstly, we describe dataset that we use for training and evaluation of our suggested methods. The second section describes the methods that we utilise for normalising the images. In the third part, we will consider the metric that will be used to evaluate the image segmentation quality. The fourth section is about the loss function that we employ to train the proposed networks. Finally, we will discuss the hyper-parameter settings to find the appropriate parameters for our proposed networks.

### 4.1 Dataset

The PROMISE12 challenge dataset [26] will be used for MRI prostate segmentation. It includes 100 T2-weighted MRI images that were collected from four different hospitals, each centre providing 25 MRI volumes, with two centres employing the EndoRectal Coil (ERC). The dataset includes 50 MRI volumes and their corresponding labels for training, and also 30 MRI volumes without ground truth images for testing. Besides, 20 unpublished MRI volumes for the live challenge. More information about the dataset is available in [Appendix A](#), Table A.1.

For the evaluation of our proposed networks, we apply ten-fold cross-validation. For each

Category	type	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF
Train	Total	957	1010	1115	1180	1167	1147	1132	1132	1137	1039
	Prostate	564	575	624	641	630	638	646	639	648	619
	ERC	418	477	679	809	809	809	761	641	569	500
Test	Total	218	202	165	97	100	110	120	125	120	120
	Prostate	101	113	90	64	73	75	65	67	72	58
	ERC	189	202	130	0	0	0	0	48	120	120
Validation	Total	202	165	97	100	110	120	125	120	120	218
	Prostate	113	90	64	73	75	65	67	72	58	101
	ERC	202	130	0	0	0	0	48	120	120	189

TABLE 4.1: This table provide the number of image slices for training, testing, and validation in the ten-fold cross-validation approach. Total, the total number of slices; Prostate, the number of slices contain prostate; ERC, the number of slices captured using EndoRectal Coil.

cross-validation fold, the training data will be separated into three categories including train, validation and test sets. We use five MRI volumes for the test, five MRI volumes for the validation and the remaining 40 volumes for the training of the suggested models in each fold. The 50 volumes have 1377 image slices. Since we are using 2D slices, Table 4.1 provides the total number of slices along with the number of slices contain prostate, and also the number of slices that were captured using the ERC. The number of slices that include prostate indicates the number of original shapes of the prostate that the network will have seen during the training; as the shapes of the prostate more vary the generalisation of the algorithm will be more increased.

Because of the limited available data, we need data augmentation to increase the number of images. Realistic data augmentation can expand the amount of data and consequently the learning capability of the network. Since we use Python Keras library [99] for implementation, the Keras data generator [100] will be used for the image augmentation. We use a collection of rotation with 10-degree range for random rotation, horizontal flip, vertical flip, zooming with the range of 10 to 12 per cent, horizontal and vertical translation, and elastic transformation [101] for augmenting the number of data to 150000 slices <sup>1</sup>.

## 4.2 Data Normalisation

One of the most critical parts of the training a CNN that can improve the learning ability of the CNN significantly is data normalisation. For the PROMISE12 dataset using a precise normalisation method can be essential to improve the quality of the final segmentation because it is collected from four different imaging centres with various imaging technologies and more importantly half

<sup>1</sup>In preliminary experiments, to find the appropriate augmented images, we compare 100000, 150000, and 200000. The results showed best performance for the total of 150000 images.

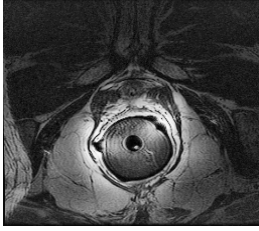

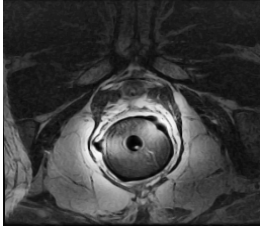
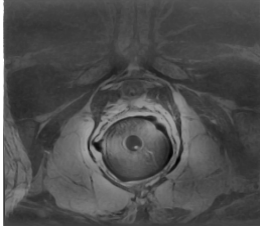

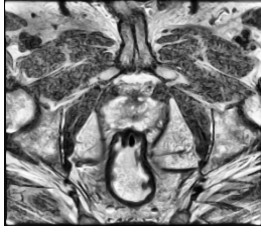
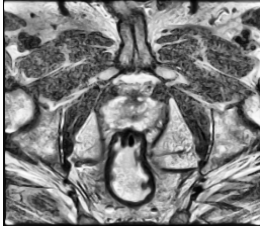
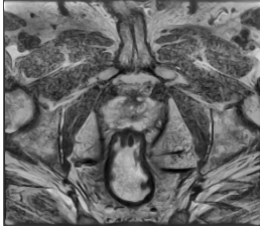
Image	Original image	NSep	NAll	NPix
ERC				
Non- ERC				

TABLE 4.2: Comparison of original ERC and non-ERC images with their corresponding normalised images. NSep, Normalise sets Separately; NAll, Normalise All; NPix, Normalise Pixels.

of those MRI images are captured using ERC.

In this research, zero mean and unit variance method (Z-score) [102] is using for data normalisation. Equation 4.1 shows the zero mean and unit variance formula, where  $x'$  is the normalised pixel value,  $\bar{x}$  is the average, and  $\sigma$  is the variance of all the pixels in the specified image slices.

$$x' = \frac{x - \bar{x}}{\sigma} \quad (4.1)$$

We utilise the zero mean and unit variance method in three ways. In the first approach, we separate the data into three groups including the training, validation and testing sets. Then we compute the mean and the variance of each group and normalise sets separately using its parameters (NSep). In the second approach, we calculate the mean and variance of all pixels in the whole set of 1377 image slices and normalise all the images using the same parameters (NAll). In the third approach, we calculate a mean and variance for each pixel position across all 1377 images and normalise the pixels at each position according to its parameters (NPix).

Table 4.2 shows two sample images, one captured using ERC and the other without ERC, together with their corresponding normalised images. As is shown, the third model seems more precise for image normalisation because its alleviate the effect of the ERC and reduces the bright regions. To find the optimum parameters, we will use the NPix method for normalising the image slices. After finding the appropriate values for all parameters, the first and the second normalisation methods will be applied to some selected models in chapter 5.

### 4.3 Evaluation Metric

For the evaluation of our proposed models, we utilise the Dice Coefficient [29–32]. The Dice Similarity Coefficient (DSC) or Sorensen Dice Coefficient is a statistical model that is usually used for comparison of a segmentation result with the ground truth image. Consider  $X$  as the predicted segmented image (set of pixels),  $|X|$  as the cardinality of  $X$ ,  $Y$  as the ground truth segmentation image, and  $|Y|$  as the cardinality of  $Y$ ; the DSC can be calculated as shown in Equation 4.2 for binary data segmentation evaluation. The output of the DSC is a number between 0 and 1. High values indicate that the obtained result and the ground truth image are alike while zero shows that they are entirely different.

$$DSC = \frac{2 |X \cap Y|}{|X| + |Y|} \quad (4.2)$$

### 4.4 Loss Function

All the machine learning methods rely on minimising or maximising a function, which is called the loss function or objective function). The loss function measures the prediction performance of our model, showing the dissimilarity between the predicted label and original label. The lower loss function value indicates, the better prediction.

In preliminary experiments, we tried different types of loss functions, and we found that in most cases using DSC loss obtained better results. Also, in the previous FCNN-based prostate image segmentation methods, DSC is used as the loss function [79–82], and they showed that it has a positive effect on the final results. Further, since we use DSC for evaluation of the proposed models, using the corresponding loss function can improve performance of the proposed models.

In our proposed networks, we employ DSC as the loss function for the training of all proposed networks (see Equation 4.2). As discussed above, DSC computes the similarity between the predicted segmentation and the ground truth image and higher value indicate the better result. Since the robustness of our proposed networks increases along with the decreasing of the loss value, the negative value of DSC employs as the loss function.

## 4.5 Hyper-parameter Setting

In this section, different parameters will be investigated. With regards that we have six different models and each model should be considered using ten-fold cross-validation, different settings will be applied only on the first three folds of the Straight and Bypass models for fine-tuning the proposed network structures. All the results in this chapter and the next chapter are based on using the cross-validation test sets for evaluation of the proposed models.

### 4.5.1 Optimiser

To find the appropriate optimiser two popular optimisers including Stochastic Gradient Descent (SGD) [103] and Adaptive Moment Estimation (ADAM) optimiser [104] will be examined for the training of our proposed networks. The SGD is a repetitive method for optimising the differentiable loss function. The SGD try to update the parameters in the inverse direction of the gradient of the loss function. Besides, ADAM optimiser is another version of SGD that can adapt the learning rate during the training using mean value and the second momentum of the gradient.

Another critical parameter that can be effective in the convergence of the algorithm and finding the (local) minimum is the learning rate which specifies the step size that the algorithm takes to reach a minimum. In other words, the objective function creates a surface and step size shows how the algorithm should follow the slope of the surface to reach a valley. We apply SGD and ADAM with various learning rates on the first three folds of the Straight and Bypass models to understand which of them is more appropriate for training of our proposed models. Firstly, we are using ADAM with the learning rate of 0.01(ADAM1), 0.001 (ADAM2), and 0.0001 (ADAM3), as well as the SGD with the same learning rates including, 0.01 (SGD1), 0.001 (SGD2), and 0.0001 (SGD3), also, we set momentum as 0.9 and the weight decay as  $1e - 6$ . Furthermore, we are applying mini batch SGD, which means after entering N images into the network, the weights of our proposed networks will be updated using backpropagation. To determine the appropriate batch size, we test four different sizes including; 8, 16, 32, and 64 on some folds of the selected models. We found that using 32 as the batch size yields better performance; therefore in all of the tests, we set batch size as 32. A part of the results provided in [Appendix A](#), Table A.2.

The proposed network architecture constructed from six blocks in the down-sampling part as well as six blocks in the up-sampling section and one block in the bottleneck along with three

Model	Fold	ADAM1	ADAM2	ADAM3	SGD1	SGD2	SGD3
Straight	Fold1	0.70	<b>0.82</b>	0.54	0.78	0.69	0.54
	Fold2	0.64	<b>0.74</b>	0.68	0.68	0.60	0.35
	Fold3	0.89	0.88	<b>0.90</b>	0.89	0.89	0.86
Bypass	Fold1	0.40	<b>0.82</b>	0.72	0.59	0.72	0.25
	Fold2	0.67	<b>0.79</b>	0.55	0.58	0.71	0.61
	Fold3	0.81	<b>0.89</b>	0.87	0.88	0.88	0.81

TABLE 4.3: Mean DSC of the Straight and Bypass models using the ADAM and SGD optimisers with different learning rates. ADAM1, learning rate is 0.01; ADAM2, learning rate is 0.001; ADAM3, learning rate is 0.0001; SGD1, learning rate is 0.01; SGD2, learning rate is 0.001; SGD3, learning rate is 0.0001.

layers in each block. To train the networks, we employ dropout at the end of each block with the probability of 0.2, and training will be continued to 25 epochs. As can be seen from Table 4.3, in both the Straight and Bypass models employing ADAM optimiser with the learning rate of 0.001 obtain better results in the first three folds on average.

The last issue for the training of the network is the number of epochs, that is one forward pass as well as one backward pass of all the training images in the network. To understand whether 25 epochs is enough for the training of our proposed models we continued training of the Bypass model with ADAM optimiser and learning rate of 0.001 and also, SGD1 and SGD2 to 45 epochs.

As is shown in Figure 4.1, during the training, training error decreased with increasing the number of epochs, but no significant change can be seen in the validation error. In conclusion, since using ADAM optimiser with the learning rate of 0.001 and batch size of 32 obtained better results and with regards that the loss value did not decrease after 25 epochs considerably, we

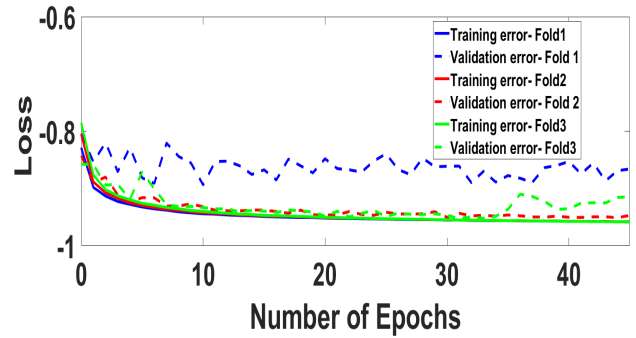


FIGURE 4.1: The training and validation error of ADAM optimiser in the Bypass model.

use ADAM with learning rate 0.001 and batch size 32 for implementation of our proposed networks. Besides, the diagram of SGD1 and SGD2 illustrate in Appendix A, Figure A.1.

#### 4.5.2 Dropout

Dropout is another important hyper-parameter of the CNN to combat overfitting and increasing the generalisation of the network. The probability of dropout and its location(s) in the network

can affect the network training capability considerably. To analyse the location and probability of dropout, we propose three different dropout configurations. In the first configuration, we locate the dropout after each layer as shown in Figure 3.2, with the probability of 0.2 (Dropout1). In the second configuration, we only employ one dropout after the last layer of each block, again with the probability of 0.2 (Dropout2). The third configuration only uses dropout after each layer in the bottleneck block, with the probability of 0.5 (Dropout3). In the preliminary experiments, to find the appropriate probability for each of the dropout configurations we examined 0.2, 0.5, and 0.8. Concerning the obtained results, the discussed dropout probabilities selected for each of the three configurations.

Model	Fold	Dropout1	Dropout2	Dropout3
Straight	Fold1	0.42	<b>0.82</b>	0.73
	Fold2	0.40	<b>0.74</b>	0.55
	Fold3	0.61	0.88	<b>0.89</b>
Bypass	Fold1	0.45	<b>0.82</b>	0.71
	Fold2	0.46	<b>0.79</b>	0.70
	Fold3	0.70	<b>0.89</b>	0.88

TABLE 4.4: Performance of using different locations and values for dropout based on mean DSC. Dropout1, drop out at the end of each layer with the probability of 0.2; Dropout2, Dropout at the end of each block with the probability of 0.2; Dropout3, dropout in the bottleneck with the probability of 0.5.

According to the obtained results of the first three folds of the Straight and Bypass models, on average the second method obtained better results in both models with 0.81, and 0.83 mean DSC respectively. It seems that using dropout after each layer can delete many units along with their connections and it can be the cause of losing more information. On the other hand, using dropout only in the bottleneck is not enough to increase the generalisation of the network. For the main experiments, we use dropout at the end of each block with the probability of 0.2. In Table 4.4 provides the detailed results of using various models of dropout.

### 4.5.3 Size of the Network

The last important parameter that is effective on the accuracy of the network is the length of the network. To find the appropriate size for our proposed network structures, firstly we use two layers per block in all the blocks. In the second configuration, three layers per block, then four



layers per block and eventually five layers per block will be used in all blocks of the proposed network structures. The first three folds of all the proposed network structures will be evaluated to find the appropriate number of layers for each of them.

As seen from Table 4.5, in the Straight model, using three layers per block obtains the best result on average with 0.81 per cent mean DSC for the first three folds. Similarly, in the Bypass model, three layers achieves the best outcome with 0.83 per cent mean DSC. Besides, The Dense model with employing three layers per block achieve 0.77 per cent mean DSC as the best performance.

For the Output from All model, five layers per block outperform the other sizes with 0.82 per cent mean DSC. Similarly, the Input to All model obtains 0.74 per cent mean DSC on average with five layers per block as the best result in the first three folds. Finally, in the Non-bypass model utilising five layers per block with 0.82 per cent mean DSC achieves the best result. To assure whether the number of layers for Output from All and Non-bypass models is enough, we further increased the number of layers per block to six, seven, and also nine.

The average mean DSC of first three folds of the Output from All model using six layers, seven layers and the nine layers per block are 0.74, 0.85, and 0.84. Besides, the Non-bypass model achieves 0.80, 0.85, and 0.83 per cent mean DSC on average using six, seven, and nine layers per block. In both network architectures, using seven layers per block outperform other sizes. The detailed results of using the different number of layers shown in [Appendix A](#), Table A.3.

So far, to find the appropriate number of blocks, we set the number of layers based on the above results and repeat the experiments with using five blocks in each part of the network as well as seven blocks. However, the experimental results demonstrate that using six blocks is more precise. The detailed results present in [Appendix A](#), Table A.4.

In summary, employing three layers for the Straight, Bypass, and Dense models, five layers for the Input to All model, and seven layers per block for the Output from All and Non-bypass models show better results in the first three folds in comparison with other numbers of layers. Most of the prior works have used two or three convolution layers per block [79, 81, 82] so far we have only examined the effect of the number of layers on the first three data folds. For a more complete comparison, all the proposed networks will be examined with two layers per block. Also, the Output from All, Input to All, and Non-bypass models will be evaluated using three layers per



Model	Fold	Two layers	Three layers	Four layers	Five layers
Straight	Fold1	0.72	<b>0.82</b>	0.72	0.73
	Fold2	0.68	<b>0.74</b>	0.46	0.46
	Fold3	0.89	0.88	<b>0.90</b>	0.87
Bypass	Fold1	0.80	<b>0.82</b>	0.81	0.81
	Fold2	0.53	<b>0.79</b>	0.72	0.72
	Fold3	<b>0.90</b>	0.89	0.89	<b>0.90</b>
Output from All	Fold1	0.82	0.81	0.83	<b>0.85</b>
	Fold2	0.73	<b>0.76</b>	0.63	0.73
	Fold3	<b>0.89</b>	<b>0.89</b>	<b>0.89</b>	<b>0.89</b>
Input to All	Fold1	0.61	<b>0.70</b>	0.59	0.58
	Fold2	0.62	0.46	0.39	<b>0.74</b>
	Fold3	0.88	0.88	0.88	<b>0.89</b>
Dense	Fold1	0.69	<b>0.79</b>	0.73	0.58
	Fold2	<b>0.65</b>	<b>0.65</b>	0.64	<b>0.65</b>
	Fold3	<b>0.89</b>	0.88	0.88	0.86
Non-bypass	Fold1	0.84	0.84	0.85	<b>0.87</b>
	Fold2	0.68	<b>0.71</b>	0.65	0.70
	Fold3	0.89	<b>0.90</b>	0.89	<b>0.90</b>

TABLE 4.5: Comparison of the obtain results of using the different number of layers in the first three folds of all proposed models.

block in all ten folds in [chapter 5](#).



# 5

## Experimental Results

Based on the preliminary experiments results, in our main experiments, there are 150000 training images after augmentation. Use the ADAM optimiser with a learning rate of 0.001, a batch size of 32, and training will continue for 25 epochs (3.25 million image presentations). Moreover, one dropout at the end of each block with the probability of 0.2 is used. Finally, in all experiments, we employ the NPix normalisation model.

To compare the six proposed networks, we run each model in all ten folds, utilising two layers and three layers per block and more layers if its effectiveness has been shown in the previous chapter. The evaluation is based on mean DSC, median DSC and the standard deviation of the DSC over the test data in each fold of cross-validation. Comparison of mean and median DSC can show the skew in the distribution of the results. The higher mean DSC, as well as median DSC, indicates the better segmentation while the lower standard deviation shows the robustness of our models. Afterwards, the effect of using different data normalisation methods and the various number of layers per block will be considered. Finally, we will analyse all the models quantitatively and qualitatively, and also we will examine the best models based on the obtained results per slice and per volume as well as considering their robustness to the EndoRectal Coil (ERC) effect.

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
S2	Mean DSC	0.72	0.68	<b>0.89</b>	<b>0.85</b>	0.70	0.90	0.86	<b>0.86</b>	<b>0.88</b>	0.87	0.821
	Median DSC	0.77	0.65	0.90	0.85	0.77	0.89	0.90	0.89	0.88	0.87	0.837
	STD-dev	0.11	0.05	0.03	0.02	0.18	0.01	0.08	0.08	0.02	0.04	0.062
S3	Mean DSC	<b>0.82</b>	<b>0.74</b>	0.88	<b>0.85</b>	<b>0.87</b>	<b>0.91</b>	<b>0.88</b>	0.84	0.86	<b>0.88</b>	<b>0.853</b>
	Median DSC	0.82	0.72	0.91	0.87	0.86	0.90	0.90	0.85	0.88	0.88	0.859
	STD-dev	0.04	0.06	0.06	0.05	0.02	0.01	0.03	0.08	0.03	0.03	0.041

TABLE 5.1: Performance of the Straight model using two layers (S2) three layers (S3) per block in all ten folds. M, Model; F, Fold.

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
B2	Mean DSC	0.80	0.53	<b>0.90</b>	0.81	0.83	<b>0.91</b>	<b>0.89</b>	<b>0.87</b>	0.86	<b>0.86</b>	0.826
	Median DSC	0.86	0.58	0.91	0.84	0.85	0.91	0.91	0.89	0.86	0.85	0.846
	STD-dev	0.11	0.16	0.02	0.09	0.07	0.01	0.04	0.06	0.03	0.03	0.062
B3	Mean DSC	<b>0.82</b>	<b>0.79</b>	0.89	<b>0.84</b>	<b>0.84</b>	0.90	<b>0.89</b>	<b>0.87</b>	<b>0.88</b>	<b>0.86</b>	<b>0.858</b>
	Median DSC	0.83	0.77	0.91	0.85	0.85	0.90	0.88	0.91	0.88	0.85	0.863
	STD-dev	0.03	0.02	0.02	0.02	0.07	0.01	0.02	0.07	0.01	0.03	0.033

TABLE 5.2: Performance of the Bypass model using two layers (B2) three layers (B3) per block in all ten folds. M, Model; F, Fold.

## 5.1 Straight Model (Baseline)

The Straight model, as the baseline model, shows the learning capability of a network without using shortcut connections. The results obtained for the Straight model utilising two layers (S2) and three layers (S3) per block are shown in Table 5.1. As can be seen, applying three layers per block improves the overall mean DSC results compared to two layers per block. The Straight model using three layers per block achieves 0.853 mean DSC. This is a good result for a model that has no shortcut connections and shows the baseline capability of FCNN in MRI prostate image segmentation.

## 5.2 Bypass Model

In the Bypass model, there is a shortcut connection that can transfer the input feature maps of the block to the output of the block. In this model, we use two layers per block (B2) and three layers per block (B3) obtaining results shown in Table 5.2. On average the Bypass model employing three layers per block achieves 0.858 mean DSC while using two layers achieve 0.826.

The comparison of the Straight and Bypass models using three layers per block shows that apart from the second fold, the mean DSC of the other folds are similar. The average mean DSC shows that using a bypass connection alone does not have a significant effect on the results.

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
O2	Mean DSC	0.82	0.73	0.89	<b>0.85</b>	0.82	0.90	0.86	0.88	<b>0.86</b>	0.85	0.846
	Median DSC	0.84	0.72	0.89	0.85	0.84	0.90	0.88	0.90	0.87	0.86	0.855
	STD-dev	0.06	0.06	0.03	0.03	0.08	0.01	0.07	0.05	0.03	0.03	0.45
O3	Mean DSC	0.81	0.76	0.89	0.84	0.84	0.89	0.88	0.87	<b>0.86</b>	0.69	0.833
	Median DSC	0.87	0.77	0.90	0.85	0.85	0.89	0.89	0.90	0.87	0.86	0.865
	STD-dev	0.09	0.06	0.03	0.05	0.04	0.01	0.03	0.05	0.03	0.25	0.064
O7	Mean DSC	<b>0.86</b>	<b>0.80</b>	<b>0.90</b>	<b>0.85</b>	<b>0.83</b>	<b>0.91</b>	<b>0.89</b>	<b>0.89</b>	0.84	<b>0.88</b>	<b>0.865</b>
	Median DSC	0.88	0.80	0.91	0.84	0.86	0.91	0.90	0.91	0.87	0.88	0.88
	STD-dev	0.04	0.02	0.04	0.06	0.01	0.03	0.07	0.02	0.02	0.02	0.03

TABLE 5.3: Performance of the Output from All model by employing two layers (O2), three layers (O3), and also seven layers (O7) per block in all ten folds. M, Model; F, Fold.

### 5.3 Output from All Model

The Output from All model has one concatenation at the end of each block to concatenate the input feature maps of the block (bypass connection) with the output feature maps of all layers in the block (gathering connections). In this case, the block is capable of sending all created feature maps within the block to the next block, and the network can learn based on various feature maps with multiple levels of spatial information. The results of the Output from All model using two layers (O2), three layers (O3), and seven layers (O7) per block shown in Table 5.3.

As seen in Table 5.3, utilising three layers per block has the worst result and using seven layers per block is the best size for the Output from All model. The results show that applying seven layers outperforms using two layers and three layers per block and accomplish 0.865 average mean DSC. Comparing the Output from All with the Bypass and Straight models shows that the combination of bypass and gathering connections outperforms the baseline and bypass connections.

### 5.4 Input to All Model

In the Input to All model, a concatenation operator after each layer collects the input feature maps of the block and also the output of the preceding layer to send them to the next layer. This process occurs after each layer in the block.

In the previous chapter, based on the first three fold results, using five layers appears more useful for this model. However, using all ten folds to study the Input to All model with two layers (I2), three layers (I3), and also five layers per block (I5) yields the results shown in Table 5.4.

Using two layers obtains the best average result with 0.819 mean DSC in comparison with two other sizes with 0.774 mean DSC. The results indicate that increasing the reuse of the input

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
I2	Mean DSC	0.61	0.62	0.88	<b>0.86</b>	<b>0.82</b>	0.89	<b>0.88</b>	<b>0.89</b>	<b>0.87</b>	<b>0.87</b>	<b>0.819</b>
	Median DSC	0.78	0.63	0.89	0.87	0.84	0.89	0.90	0.90	0.88	0.88	0.846
	STD-dev	0.27	0.07	0.03	0.01	0.07	0.02	0.03	0.05	0.02	0.04	0.061
I3	Mean DSC	<b>0.70</b>	0.46	0.88	0.84	0.66	<b>0.90</b>	0.84	0.83	0.79	0.84	0.774
	Median DSC	0.82	0.52	0.88	0.86	0.80	0.89	0.82	0.85	0.79	0.83	0.806
	STD-dev	0.23	0.25	0.02	0.04	0.24	0.02	0.04	0.07	0.06	0.03	0.1
I5	Mean DSC	0.58	<b>0.74</b>	<b>0.89</b>	0.81	0.78	0.83	0.82	0.82	0.62	0.85	0.774
	Median DSC	0.62	0.65	0.90	0.84	0.84	0.83	0.83	0.80	0.73	0.88	0.792
	STD-dev	0.28	0.26	0.03	0.05	0.14	0.04	0.04	0.05	0.31	0.04	0.124

TABLE 5.4: Performance of the Input to All model using two layers (I2), three layers (I3), and five layers (I5) per block in all ten folds. M, Model, F, Fold.

feature maps within the block (scatter connections) not only has not improved the accuracy of segmentation but it has actually decreased the efficiency of segmentation considerably. Comparing the baseline Straight model with the Input to All model shows that a network without shortcut connections can sometimes outperform a network with the wrong or poor shortcut connections.

## 5.5 Dense Model

In the Dense model, after each layer, the concatenation collects all the output feature maps of the preceding layers in the block along with the input feature maps of the block and feeds them to the next layer. The obtained results of applying the Dense model for the MRI prostate segmentation using two layers (D2) and three layers (D3) are presented in Table 5.5. As is shown in the table, except for the first fold, the results of the other folds are similar. Overall, employing three layers per block with the average mean DSC of 0.834 outperforms the Dense model using two layers. Given that this model is applying all possible shortcut connections in each block (bypass, gathering, scatter, and internal connections), it should be capable of outperforming all the other proposed models. However, the results only exceed the results of the Input to All model (which uses only bypass and scatter connections).

## 5.6 Non-bypass Model

The results obtained by the Non-bypass model are shown in Table 5.6 for all ten-fold using two layers (NB2), three layers (NB3), and also seven layers (NB7) per block. There is no significant difference between employing two layers or three layers in this model, however by increasing the number of layers to seven per block, the mean DSC improves to 0.873. In this model, we omit the connection between the input and the output of the block that consequently decreases the number

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
D2	Mean DSC	0.69	<b>0.65</b>	<b>0.89</b>	0.81	0.81	<b>0.91</b>	<b>0.88</b>	0.88	0.84	<b>0.86</b>	0.822
	Median DSC	0.81	0.59	0.90	0.86	0.87	0.91	0.89	0.91	0.85	0.87	0.846
	STD-dev	0.30	0.12	0.03	0.09	0.12	0.01	0.03	0.05	0.02	0.03	0.08
D3	Mean DSC	<b>0.79</b>	<b>0.65</b>	0.88	<b>0.83</b>	<b>0.82</b>	0.89	0.87	<b>0.89</b>	<b>0.86</b>	<b>0.86</b>	<b>0.834</b>
	Median DSC	0.87	0.65	0.87	0.87	0.83	0.89	0.88	0.91	0.87	0.88	0.852
	STD-dev	0.11	0.09	0.03	0.07	0.07	0.02	0.03	0.03	0.02	0.04	0.051

TABLE 5.5: Performance of the Dense model by employing two layers (D2), and three layers (D3) per block in all ten folds. M, Model; F, Fold.

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
NB2	Mean DSC	0.84	0.68	0.89	0.85	0.82	0.90	0.88	0.88	0.85	0.87	0.846
	Median DSC	0.86	0.67	0.90	0.87	0.86	0.91	0.88	0.91	0.84	0.87	0.857
	STD-dev	0.04	0.14	0.03	0.04	0.10	0.01	0.03	0.06	0.03	0.03	0.051
NB3	Mean DSC	0.84	0.71	<b>0.90</b>	<b>0.86</b>	0.77	0.90	0.89	0.88	0.87	0.86	0.848
	Median DSC	0.87	0.69	0.91	0.87	0.83	0.90	0.90	0.89	0.86	0.87	0.859
	STD-dev	0.04	0.06	0.03	0.03	0.19	0.02	0.03	0.05	0.02	0.03	0.05
NP7	Mean DSC	<b>0.87</b>	<b>0.78</b>	<b>0.90</b>	<b>0.86</b>	<b>0.85</b>	<b>0.92</b>	<b>0.90</b>	<b>0.89</b>	<b>0.88</b>	<b>0.88</b>	<b>0.873</b>
	Median DSC	0.88	0.78	0.91	0.87	0.85	0.91	0.89	0.90	0.88	0.88	0.88
	STD-dev	0.03	0.07	0.03	0.04	0.07	0.01	0.02	0.05	0.02	0.02	0.03

TABLE 5.6: Performance of the Non-bypass model by applying two layers (NB2), three layers (NB3), and seven layers (NB7) per block in all ten folds. M, Model; F, Fold.

of parameters and also, increases the accuracy of the network.

## 5.7 Comparison of Normalisation Methods

As is discussed in [chapter 4](#), we employ the zero mean and unit variance normalisation method in three ways. The results of applying the NSep and the NAll normalisation methods on the Output from All model can be seen in Table 5.7. The NSep normalisation method obtains 0.809 mean DSC while the NAll normalisation method achieves 0.838 mean DSC. The results show that the NPix method, which uses the average image and variance image for normalisation, is more precise for segmentation and obtains 0.865 mean DSC. Similar experiments show that the NSep and NAll normalisation methods also decrease the accuracy of segmentation in the Non-bypass model. See Table A.5 in [Appendix A](#).

## 5.8 Analysing Unequal Layers per Block

As another variation, we considered the Output from All and Non-bypass networks where the number of layers per block increases closer to the bottleneck. We employ the following patterns of

M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
NSep	Mean DSC	0.72	0.64	0.88	0.75	0.73	0.88	0.88	0.88	0.86	0.87	0.809
	Median DSC	0.82	0.65	0.88	0.83	0.84	0.88	0.88	0.90	0.87	0.88	0.843
	STD-dev	0.15	0.12	0.03	0.13	0.2	0.02	0.03	0.04	0.03	0.04	0.158
NAll	Mean DSC	0.78	0.67	<b>0.90</b>	<b>0.85</b>	0.79	<b>0.91</b>	0.87	0.87	<b>0.87</b>	0.87	0.838
	Median DSC	0.86	0.71	0.89	0.85	0.87	0.90	0.87	0.90	0.87	0.88	0.86
	STD-dev	0.13	0.14	0.02	0.04	0.15	0.01	0.04	0.07	0.03	0.02	0.06
NPix	Mean DSC	<b>0.86</b>	<b>0.80</b>	<b>0.90</b>	<b>0.85</b>	<b>0.83</b>	<b>0.91</b>	<b>0.89</b>	<b>0.89</b>	0.84	<b>0.88</b>	<b>0.865</b>
	Median DSC	0.88	0.80	0.91	0.84	0.86	0.91	0.90	0.91	0.87	0.88	0.88
	STD-dev	0.04	0.02	0.04	0.06	0.01	0.03	0.07	0.02	0.02	0.02	0.03

TABLE 5.7: Performance of the three normalisation methods on the Output from All model. NSep, Normalise sets Separately; NAll, Normalise All; NPix, Normalise Pixels M, Model; F, Fold.

block sizes <sup>1</sup>, 3-3-5-7-9-11-13, 4-4-6-6-8-8-12, 5-5-7-9-9-11-13, 4-4-6-8-10-12-14, 4-4-5-7-10-12-15. The results show that in most cases, employing an equal number of layers outperforms an unequal number of layers per block. However, in some cases, the results of the two approaches are quite similar. For example, using the 4-4-5-7-10-12-15 pattern for the Output from All and Non-bypass models, obtain 0.84 mean DSC. The results are presented in [Appendix A](#), Table A.6.

## 5.9 Quantitative Comparison

The Input to All model is the worst, and the Output from All and Non-bypass models are the best models among all six proposed networks for prostate MRI segmentation (see Table 5.8). The Non-bypass model outperforms all other models with 0.873 mean DSC.

The results of the Straight model show that it is possible a network without using shortcut connections outperforms

Method	Mean DSC	Median DSC	STD-dev
Mun et al. [82]	0.853	-	-
Yu et al. [81]	0.869	-	-
Straight	0.853	0.859	0.41
Bypass	0.858	0.863	0.033
Output from All	0.865	0.88	0.03
Input to All	0.819	0.846	0.061
Dense	0.834	0.852	0.051
Non-bypass	<b>0.873</b>	<b>0.88</b>	<b>0.03</b>

TABLE 5.8: Quantitative comparison of proposed models with another model.

networks using different patterns of shortcut connections. For the Bypass model, the results show that using the bypass connection does not have a significant effect on the final segmentation. In the Output from All model, the average mean DSC increases to 0.865 using bypass and gathering connections. For the Input to All model, where the input feature maps of the block are reused

<sup>1</sup>Each pattern shows the number of layers of the down-sampling and the bottleneck. For example, the first full pattern is 3-3-5-7-9-11-13-11-9-7-5-3-3.



several times within the block, the results show reduced segmentation performance. In fact, this is the worst performance among all six proposed models. In the Dense model, each layer employs all possible features including the input feature maps of the block as well as the output of all previous layers in the block. However, this model could not even compete with the Straight model. Finally, the Non-bypass model by omitting input to output connection and decreasing the number of feature maps in comparison with the Dense model improved the results to 0.873 which is the best result. Since omitting the bypass connection in the Non-bypass model, improved the results considerably against the Dense model, a Non-bypass Input to All model and Non-bypass Output from All model are two models that may improve the segmentation results and could be investigated in the future.

As discussed in [chapter 3](#), the number of feature maps that move between blocks in the Bypass and Input to All models, and also, in the Output from All and Dense models, are the same. However, the results indicate that the Bypass and Output from All models which employ fewer feature maps within the blocks achieve better results. It shows that, in addition to the number of feature maps that transfer between blocks, the number of feature maps that are created and move inside the blocks also affect the final results. Furthermore, comparison of the Straight, Bypass and Output from All models that have similar internal structures indicate that the Output from All model that transfer more diverse features to the next block is more precise in comparison to the Straight and Bypass models. The results demonstrate that finding a precise structure is a trade-off between internal and external block structures.

We use, the Wilcoxon signed rank test [\[105\]](#) to show the statistically significant differences among the proposed approaches. We compare the Non-bypass model with all other models (see [Table 5.9](#)), and the statistical comparison of the mean DSC over all ten folds shows that the improvement of the Non-bypass model is statistically significant ( $p < 0.05$ ) in comparison with all other proposed networks except the Output from All model. The Output from All model is our the second-best proposed network.

Method	$p < 0.05$
Straight	<b>0.0137</b>
Bypass	<b>0.0137</b>
Output from All	0.0645
Input to All	<b>0.0020</b>
Dense	<b>0.0039</b>

TABLE 5.9: Statistically significant comparison results using the Wilcoxon signed rank test.

Another important feature that can affect the training of the network is the number of parameters, which we show in Table 5.10. As can be seen, the Straight model has the least, and the Dense model has the most parameters using three layers per block. However, based on the optimum number of layers per block, increasing the number of layers to seven in the Output from All and Non-bypass models increases the number of parameters considerably.

Whereas the Input to All model employs two layers, other models use three or more layers per block as the optimum. Although the output from All and Non-bypass models using seven layers have the most parameters, they achieve the best results. It shows that besides the number of parameters, the type of feature maps is also important.

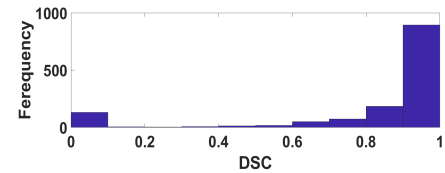
Histogram of the results of the Output from All and Non-bypass models across all ten folds (1377 slices) are shown in Figure 5.1. As seen, some segmented images have low mean DSC between 0 to 0.1 in both networks. These images primarily contain small prostate regions that cannot be approximated by 2D networks.

The number of images with mean DSC between 0.8 and 1 in the Non-bypass model is more than the Output from All model. Figures A.2 and Figure A.3 show the results of all ten folds separately in Appendix A.

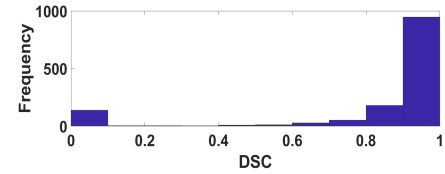
Also, to show the performance of the best and second-best proposed models for the segmentation of the prostate with the different sizes we provide figure 5.2. Firstly, we analyse the size of the prostate on all 1377 image slices based on

Method	LPB=3	LPB=opt
Straight	12,604,377	12,604,377
Bypass	17,194,250	17,194,250
Output from All	33,414,442	81,612,330
Input to All	28,011,242	18,663,370
Dense	66,597,514	66,597,514
Non-bypass	40,123,513	214,709,881

TABLE 5.10: The number of parameters of proposed networks. LPB, Layer Per Block; opt, Optimum number of layer.

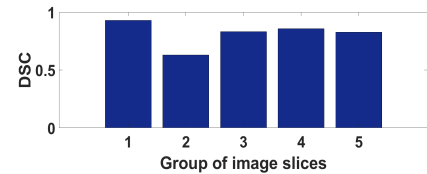


(a) The Output from All model.

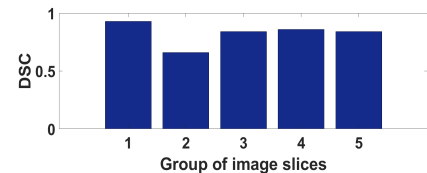


(b) The Non-bypass model.

FIGURE 5.1: Comparison of the Output from All and Non-bypass models based on all MRI slices.



(a) The Output from All model.



(b) The Non-bypass model.

FIGURE 5.2: Comparison of the Output from All and Non-bypass models based on the size of the prostate.

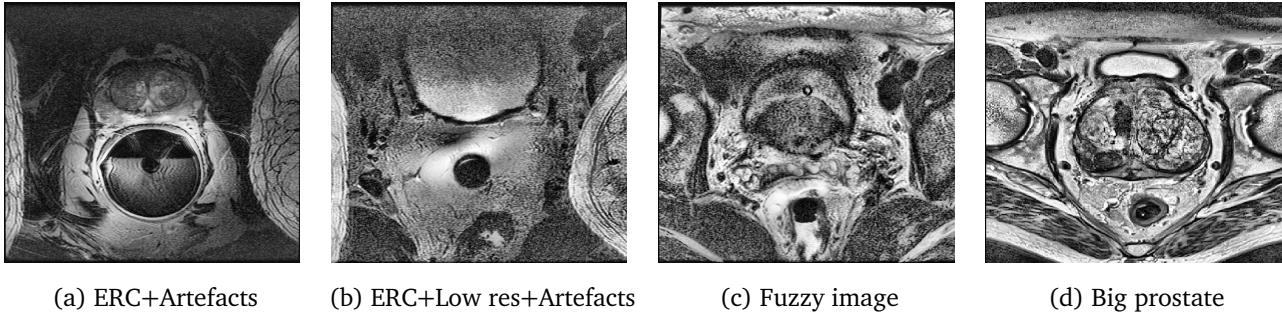


FIGURE 5.3: Four sample images to show the quality of images.

the number of pixels of the prostate. Among 1377 image slices only 788 image slices contain prostate. The smallest prostate has 418 pixels, the largest has 4625 pixels, and in average the prostate contains 2315 pixels in the dataset. We divide the images into five groups, the first group contain images without prostate (599 images), the second group contains images with 418 to 1366 prostate pixels (109 images), the third group contains images with 1367 to 2315 prostate pixels (242 images), the fourth group contains the images with 2316 to 3470 prostate pixels (352 images), and finally the fifth group contains the images that have 3471 to 4625 prostate pixels (75 images). As can be seen from figure 5.2, images with small prostate have lower DSC than other images. The Non-bypass proposed model obtain better results than the Output from All model for segmentation of the image slices with small prostate.

## 5.10 Analysis of Data Folds

Performance results show that the first, second, and the fifth folds are the most difficult folds for segmentation, such that in some cases their results change the ranking of the proposed models. However, for other folds, the segmentation results using different methods are much similar. These folds mainly contain the high-resolution test images.

The test set of the first fold contains five MRI volumes of which four were captured using the ERC and have bright regions around the ERC similar to Figures 5.4a and 5.4b. Also, some of the images have low contrast and wrap-around artefacts (see Figures 5.4a and 5.4b) that makes the prostate segmentation even more challenging. In the second fold, all the images in the test set were captured using the ERC and also, the images have poor contrast resolution (everything is dark or bright), low spatial resolution (fuzzy images—see Figure 5.4c), and contain wrap-around artefacts in most of the images that make this fold the most challenging fold for segmentation. Furthermore, in the entire dataset, there is only one MRI volume that contains a very large prostate

(see Figure 5.4d)—that volume appears in the fifth fold test set. Since the network for this fold has never seen such a large prostate during training, most of the models could not segment this prostate precisely despite the clear and high-resolution image data. In contrast, the sixth fold yields the best segmentation performance because the test images are non-ERC and have good spatial and contrast resolutions. The test sets of the ninth and tenth folds are all captured using the rectal coil, but the images have high resolution and a large field of view that allows the network to use landmarks to find the prostate easily.

The results show that for segmentation of MRI slices with good spatial and contrast resolutions there is no significant difference among well-structured models. The problematic folds highlight the actual differences among the proposed models.

## 5.11 Comparison with Prior Work

In prior work, only ten FCNN-based prostate segmentation methods have been published in conference proceedings or journals. Zhu et al. [79] utilised their unpublished dataset achieving 0.885 DSC. However, they excluded non-prostate slices which improves the DSC of their 2D network by reducing spurious detections. Also, Ji et al. [106] and Clark et al. [107], excluded non-prostate slices from the PROMISE12 dataset and obtained 0.91 and 0.86 mean DSC using ten-fold and four-fold cross-validation respectively. Milletari et al. [80], Chen et al. [108], and Liu et al. [109] evaluated their networks on the test set of the PROMISE12 dataset for which ground truth is not publicly available and obtained 0.869, 0.895, and 0.86 mean DSC respectively. In addition, Drozdal et al. [110], used two FCNN for segmentation. Firstly, they segmented the input image using FCNN and then used another residual based FCNN for boundary refinement. They applied their proposed framework for different organ MRI segmentation including prostate MRI segmentation and obtained 0.874 mean DSC on the test set of the PROMISE12 dataset. Finally, Sun et al. [111], proposed an interactive framework for medical image segmentation and a part of this framework is FCNN. They obtained 89.81 mean DSC on the test set of PROMISE12 dataset. Neither of these results are comparable with our results because they use different test conditions.

Yu et al. [81] evaluated their proposed model on both the PROMISE12 training set using cross-validation, achieving 0.869 DSC. They also report 0.894 DSC on the test set on the PROMISE12 dataset, which is not comparable with our models. Finally, Mun et al. [82] tested their 3D-FCNN

(BCNN) network using ten-fold cross-validation on the training set of the PROMISE12 dataset and achieved 0.853 mean DSC. We compare our models with the cross-validation results in these two papers (see Table 5.8). Given that 3D methods segment the prostate as a 3D volume (they use information from adjacent slices), finding the prostate will be more straightforward and precise than the 2D models, especially in the lower and upper slices of the prostate volumes where the prostate is a small part of the image. However, as is shown in Table 5.8, the Non-bypass 2D network outperforms both 3D methods for prostate image segmentation and achieves new state-of-the-art FCNN-based prostate segmentation results.

## 5.12 Qualitative Comparison

As a subjective evaluation of the Non-bypass model as the best model, six images selected from the test set of the different folds and the segmentation results are presented in Figure 5.4 here the red border shows the ground truth and the green border indicates the predicted border. As can be seen, five out of six images were captured using ERC (the sixth image is non-ERC). The first and second images (see Subfig 5.4a and Subfig 5.4b), were captured using ERC and the bright region can be seen around the rectal coil. However, the NPix normalisation method compensates the bright region and our proposed model segments the prostate properly. Also, the Non-bypass model segmented the prostate precisely in the third image (see Subfig 5.4c), despite the wrap-around artefacts in the image. The fourth and fifth images (see Subfig 5.4d and Subfig 5.4e), contain only small regions of the prostate, but our proposed method still segments the prostate accurately and specifically in the fourth image the rectal coil effect is similar in to size the prostate. The last image (see Subfig 5.4f) is the non-ERC image that is also segmented correctly by the Non-bypass model. Overall, the results show the capability of our best model in the segmentation of the prostate MRI. A subset of the results of all proposed models in five different folds are presented in Appendix A, Table A.7, Table A.8, Table A.9, Table A.10, and Table A.11.

## 5.13 Analysing the EndoRectal Coil Effect

The PROMISE12 dataset includes 50 MRI volumes for training with 24 of them captured using the ERC. These volumes include 809 image slices while the 26 non-ERC volumes include only 568 slices. To show the effect of ERC on the final segmentation results, we evaluate the results of the



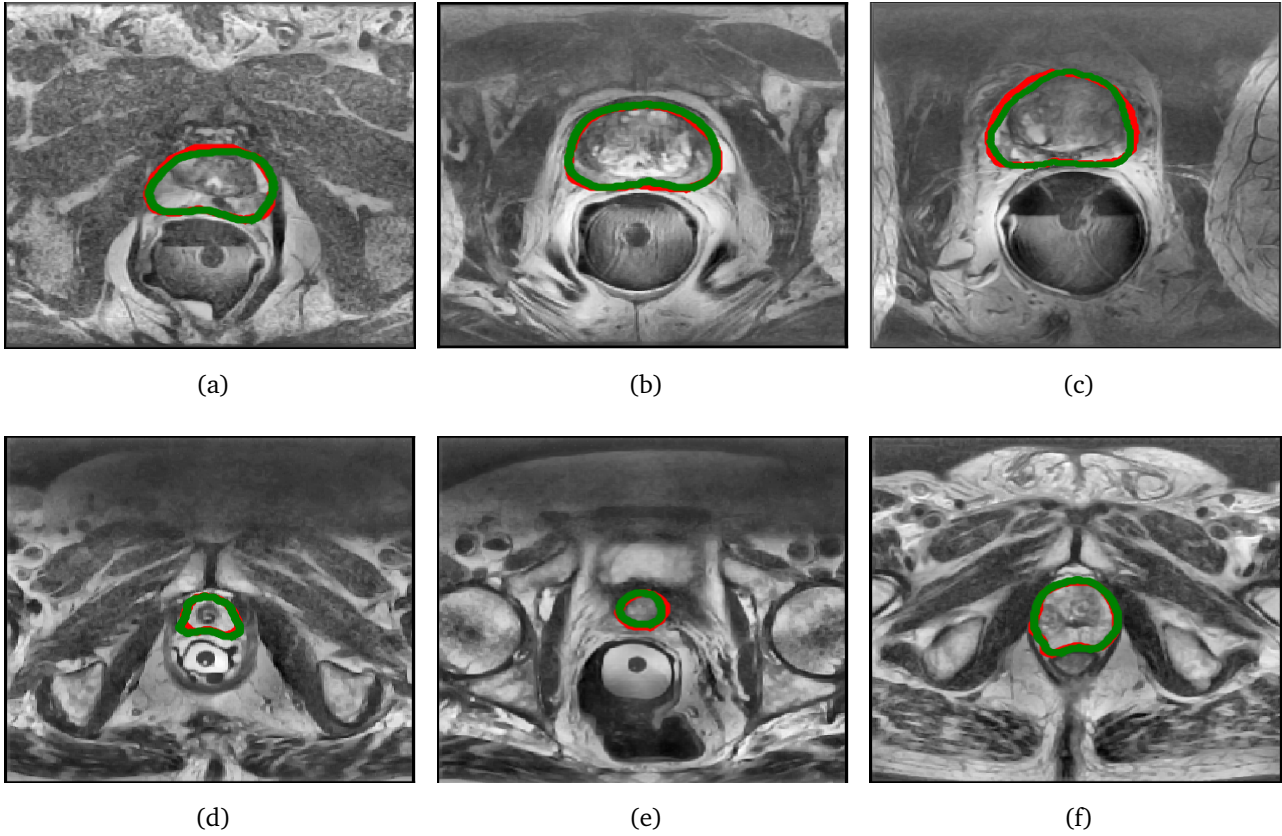


FIGURE 5.4: The six sample segmented images using Non-bypass model. The red border is the ground truth and the green border in the predicted border.

Output from All and Non-bypass models based on the obtained DSC per volume.

The obtained results of the Output from All model per volume is shown in Figure 5.5a where the red bars indicate the DSC of the ERC volumes and the blue bars show the DSC of the non-ERC volumes. In the Output from All model, the average mean DSC of ERC volumes is 0.8576 and the average DSC of non-ERC volumes is 0.8727. The average DSC in the non-ERC volumes is higher than ERC volumes but, in this model, the best individual result is achieved by segmentation of an ERC volume, and segmentation of a non-ERC volume produces the worst individual outcome.

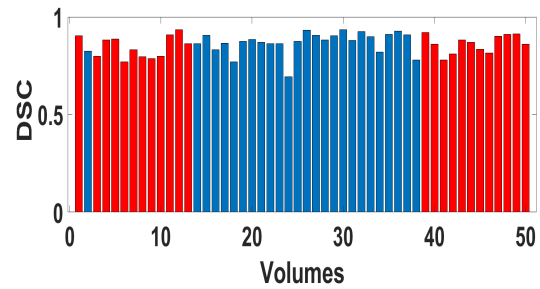
In Figure 5.5b the results of the Non-bypass model per volume are presented. In this model, the average DSC of ERC volumes is 0.8698 and the average DSC of non-ERC volumes is 0.8749. The Non-bypass model was more precise in the segmentation of both ERC and Non-ERC volumes, and more importantly we can see the smaller differences between average mean DSC of the ERC and Non-ERC volumes.

To demonstrate there are no statistically significant differences between the segmentation of the ERC and non-ERC volumes using the Output from All and Non-bypass models, we evaluate

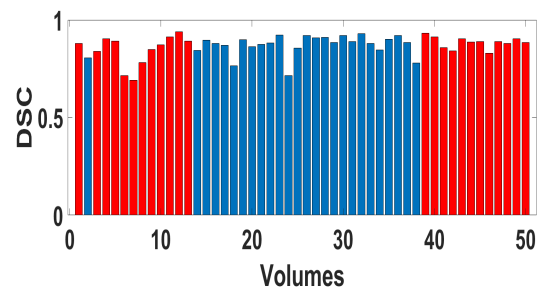
the results using Wilcoxon–Mann–Whitney test [112], which calculate the differences between two independent sets. The comparison of the obtained results of the Output from All model, for segmentation of the ERC and non-ERC volumes using Wilcoxon–Mann–Whitney test is  $p=0.26$ , and for the Non-bypass model  $p=0.90$ . These results demonstrate that there are no significant differences between the segmentation of the ERC and Non-ERC volumes either model.

These test results show that using ERC is not the only reason for a low mean DSC. The most important reasons are the low contrast and spatial resolutions and artefacts that can decrease the accuracy of the segmentation, particularly for the first and the second folds. Also, low diversity of the training images in the fifth fold. In both models, the lowest mean DSC is for the 24th volume that is a non-ERC and contains the only example of the large prostate. Using zoom augmentation could address this problem somewhat, however increasing the zoom factor is not an appropriate solution because the features in the field of view around the prostate (muscle, fat, etc.) will be lost.

In summary, we have investigated the performance of all six propose networks in detail concerning, slices, volumes, and cross-validation folds. The results demonstrate that the Non-bypass model is the most precise model among all our recommended models. The results show that using a bypass connection alone is ineffective and when used along with the scatter connections obtains the worst results. Also, we found that the resolution of the images is a critical factor for the segmentation since the ERC volumes with good resolution are segmented precisely.



(a) The Output from All model results per volume.



(b) The Non-bypass model results per volume.

FIGURE 5.5: Comparison of the Output from All and Non-bypass models based on obtain results per volume. Red bars, ERC-volumes; Blue bars, non-ERC volumes.





## Conclusion and Future Work

In this thesis, we propose six FCNN-based network structures for MRI prostate segmentation. Our proposed Non-bypass model outperforms the comparable 3D FCNN-based segmentation methods when evaluated by cross-validation on the PROMISE12 training dataset, and achieves a new state-of-the-art for FCNN prostate segmentation on the PROMISE12 training dataset.

In our research, we have analysed different parameters of FCNN with a particular focus on using the shortcut connection. The results of our novel structures show the benefits and advantages of reusing the extracted feature maps within and between the blocks, and also the impact of the network structure on the prostate MRI segmentation. Shortcut connections can help their network. However, our results show that using shortcut connections can also decrease the accuracy of the network; therefore, it is critical to use shortcut connections in the proper place in the network. Our experiments show that the bypass connection, which transfers the input feature maps to the output of the block is not beneficial for the prostate segmentation and when it is used together with scattering connections it even significantly reduces performance. In contrast, gathering connections, which collect the output feature maps of the layers can significantly improve performance in this application. Also, the results show that, among the models that transfer the

equal number of feature maps between the layers, the models with the simpler architecture within the block produce better results.

The Straight (Baseline) model which uses no shortcut connections, achieves competitive results. The Bypass model demonstrated that transferring the input feature maps of the current block to the next block has no significant benefit for the final segmentation results compared the baseline. The results of the Input to All model and the Dense model (two other example of using bypass connection along with other connections) show that shortcut connections sometimes decrease the accuracy of the prostate segmentation in comparison with Baseline network. The Output from All model and the Non-bypass model are the most precise models among all six proposed models. The results demonstrated that a well-structured model could segment both ERC and Non-ERC images precisely and the effect of using different patterns of shortcut connections is more evident in the difficult folds.

It is necessary to emphasise that, for training our proposed 2D networks we use the entire images without cropping, and the results are obtained without any post-processing for boundary refinement. Moreover, based on our experiments in this project, batch normalisation and data augmentation play essential roles in the training of the network. In the case of the only large prostate, our models could not segment the images accurately because they have not seen a similar image during training. A more sophisticated augmentation method could possibly address this problem. Also, some of the MRI images suffer from low resolution and artefacts that can be addressed with improved normalisation methods and preprocessing.

In the future work, based on the obtained results, we plan to test Non-bypass Output from All and Non-bypass Input to all models to analyse the effect of the bypass connection on the discussed models. Also, we plan to apply our models to MRI datasets of other organs. Although this thesis is focused specifically on segmentation of the prostate in MR images, the methods that we have developed could equally be applied to other segmentation tasks, including other medical applications. Further, we plan to convert our proposed 2D models into 3D models, which can better segment small prostate regions. Since the available training data for prostate segmentation is limited; another opportunity would be to use transfer learning [113] for MRI prostate segmentation. This would include fine-tuning a pre-trained network and employing it for prostate segmentation. Furthermore, adding an attention mechanism [114] in our proposed models may improve the

results. Attention mechanisms direct the network to focus on the important region in the image. Finally, we intend to extend our work to prostate cancer detection...





## Appendix: Detailed Results

---

**Algorithm 1** Batch Normalisation transform on mini-batch.

---

**Input:** Values of  $x$  over a mini-batch  $B$ .

**Output:**  $y_i = BN_{\gamma, \beta}(X_i)$

- 1: **function** BATCH NORMALISATION(  $B = x_{1, \dots, m}$  )
  - 2:      $\mu_B \leftarrow \frac{1}{m} \sum_{i=1}^m x_i$                                  //mini-batch mean
  - 3:      $\sigma_B^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$                  //mini-batch variance
  - 4:      $\hat{x}_i \leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$                                  //normalise
  - 5:      $y_i \leftarrow \gamma \hat{x}_i + \beta \equiv BN_{\gamma, \beta}(x_i)$                  //scale and shift
  - 6: **end function**
-

Center	Field strength	ERC	Resolution	Imaging device
HUH	1.5	Y	0.625/3.6	Siemens
BIDMC	3	Y	0.25/2.2 – 3	GE
UCL	1.5/3	N	0.325 – 0.625/3 – 3.6	Siemens
RUNMC	3	N	0.5 – 0.75/3.6 – 0.4	Siemens

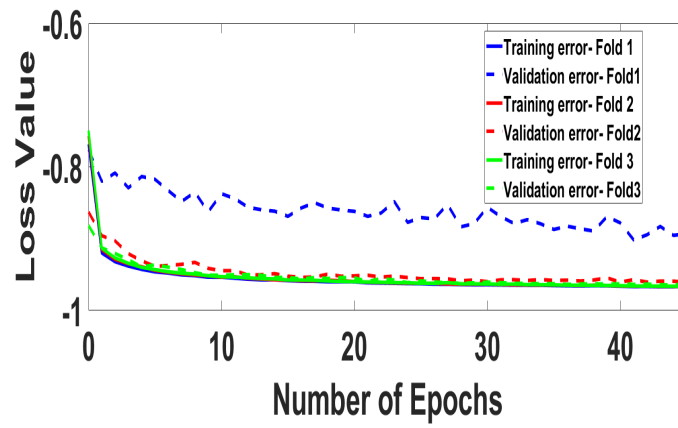
TABLE A.1: This table provide the Centers name, Filed strength, Resolution, and Imaging devices that used for collecting promise dataset. HUH, Haukeland University Hospital; BIDMC, Beth Israel Deaconess Medical Center; UCL, University College London; RUNMC, Radboud University Nijmegen Medical Center; ERC, EndoRectal Coil.

Model	Fold	Batch1	Batch2	Batch3	Batch4
Straight	Fold1	0.59	0.72	<b>0.82</b>	0.51
	Fold2	0.52	0.64	<b>0.74</b>	0.45
	Fold3	0.68	0.78	<b>0.88</b>	0.68
Bypass	Fold1	0.65	0.73	<b>0.82</b>	0.53
	Fold2	0.53	0.63	<b>0.79</b>	0.50
	Fold3	0.70	0.79	<b>0.89</b>	0.73

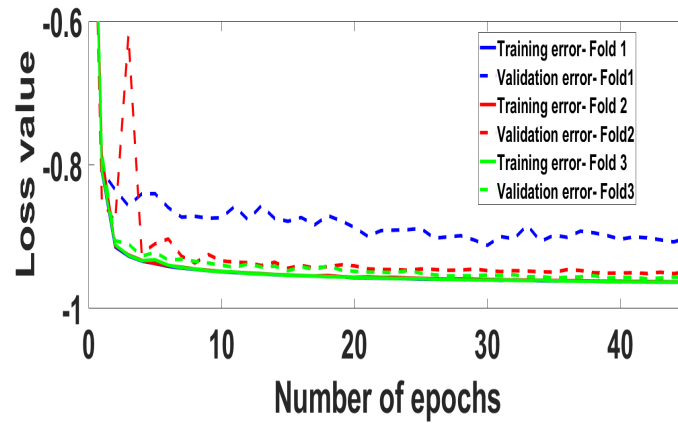
TABLE A.2: Comparison of using different values for the batch size based on mean DSC. Batch1, ADAM optimiser, learning rate 0.001, batch size 8; Batch2, ADAM optimiser, learning rate 0.001, batch size 16; Batch3, ADAM optimiser, learning rate 0.001, batch size 32; Batch4, ADAM optimiser, learning rate 0.001, batch size 64.

Model	Fold	2 layers	3 layers	4 layers	5 layers	6 layers	7 layers	9 layers
Straight	Fold1	0.72	<b>0.82</b>	0.72	0.73	-	-	-
	Fold2	0.68	<b>0.74</b>	0.46	0.46	-	-	-
	Fold3	0.89	<b>0.88</b>	<b>0.90</b>	0.87	-	-	-
Bypass	Fold1	0.80	<b>0.82</b>	0.81	0.81	-	-	-
	Fold2	0.53	<b>0.79</b>	0.72	0.72	-	-	-
	Fold3	<b>0.90</b>	0.89	0.89	<b>0.90</b>	-	-	-
Output from All	Fold1	0.82	0.81	0.83	0.85	0.67	<b>0.86</b>	0.84
	Fold2	0.73	0.76	0.63	0.73	0.68	<b>0.80</b>	<b>0.81</b>
	Fold3	0.89	0.89	0.89	0.89	0.88	<b>0.90</b>	0.89
Input to All	Fold1	0.61	<b>0.70</b>	0.59	0.58	-	-	-
	Fold2	0.62	0.46	0.39	<b>0.74</b>	-	-	-
	Fold3	0.88	0.88	0.88	<b>0.89</b>	-	-	-
Dense	Fold1	0.69	<b>0.79</b>	0.73	0.58	-	-	-
	Fold2	<b>0.65</b>	<b>0.65</b>	0.64	<b>0.65</b>	-	-	-
	Fold3	<b>0.89</b>	0.88	0.88	0.86	-	-	-
Non-bypass	Fold1	0.84	0.84	0.85	<b>0.87</b>	<b>0.87</b>	<b>0.87</b>	0.86
	Fold2	0.68	0.71	0.65	0.70	0.65	<b>0.78</b>	0.75
	Fold3	0.89	<b>0.90</b>	0.89	<b>0.90</b>	0.89	<b>0.90</b>	0.89

TABLE A.3: Performance of using the different number of layers (two to nine) per block in the first three folds of all proposed models.



(a) SGD-Learning rate 0.01



(b) SGD-Learning rate 0.001

FIGURE A.1: Comparison of loss error using different optimisation methods for the Bypass model.

Model	Fold	Five blocks	Six blocks	Seven blocks
Straight	Fold1	0.81	<b>0.82</b>	0.79
	Fold2	0.53	<b>0.74</b>	0.65
	Fold3	<b>0.89</b>	0.88	0.88
Bypass	Fold1	<b>0.83</b>	0.82	<b>0.83</b>
	Fold2	0.56	<b>0.79</b>	0.59
	Fold3	<b>0.89</b>	<b>0.89</b>	<b>0.89</b>
Output from All	Fold1	<b>0.86</b>	<b>0.86</b>	0.85
	Fold2	0.70	<b>0.80</b>	0.73
	Fold3	0.89	<b>0.90</b>	0.89
Input to All	Fold1	<b>0.70</b>	0.58	0.23
	Fold2	0.62	<b>0.74</b>	0.48
	Fold3	<b>0.89</b>	<b>0.89</b>	0.74
Dense	Fold1	0.70	<b>0.79</b>	0.68
	Fold2	0.60	<b>0.65</b>	0.57
	Fold3	<b>0.89</b>	0.88	0.84
Non-bypass	Fold1	<b>0.87</b>	<b>0.87</b>	0.85
	Fold2	0.73	<b>0.78</b>	0.70
	Fold3	0.89	<b>0.90</b>	0.88

TABLE A.4: Performance of using different number of blocks in the down-sampling and the up-sampling parts in the all proposed models .

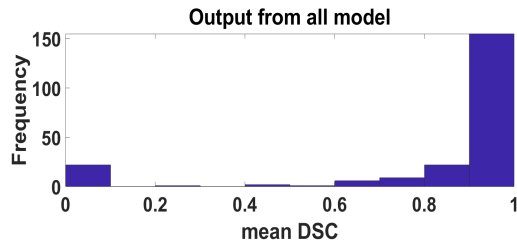
M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
NSep	Mean DSC	0.79	0.72	0.88	0.83	0.76	0.88	0.88	0.88	0.87	0.87	0.836
	Median DSC	0.84	0.69	0.88	0.88	0.84	0.87	0.87	0.90	0.87	0.88	0.852
	STD-dev	0.12	0.06	0.03	0.08	0.2	0.03	0.02	0.06	0.02	0.02	0.064
NAll	Mean DSC	0.86	0.77	0.89	0.85	0.84	0.91	0.86	0.88	<b>0.88</b>	0.86	0.86
	Median DSC	0.87	0.78	0.90	0.85	0.86	0.90	0.89	0.89	0.88	0.87	0.869
	STD-dev	0.04	0.07	0.03	0.04	0.07	0.01	0.07	0.04	0.03	0.03	0.043
NPix	Mean DSC	<b>0.87</b>	<b>0.78</b>	<b>0.90</b>	<b>0.86</b>	<b>0.85</b>	<b>0.92</b>	<b>0.90</b>	<b>0.89</b>	<b>0.88</b>	<b>0.88</b>	<b>0.873</b>
	Median DSC	0.88	0.78	0.91	0.87	0.85	0.91	0.89	0.90	0.88	0.88	0.88
	STD-dev	0.03	0.07	0.03	0.04	0.07	0.01	0.02	0.05	0.02	0.02	0.03

TABLE A.5: Performance of the three normalisation methods on the Non-bypass model. NSep, Normalise sets Separately; NAll, Normalise All; NPix, Normalise Pixels M, Model; F, Fold.

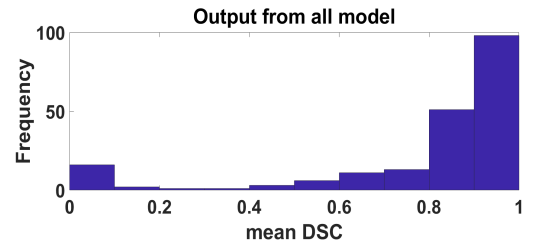
M	Criteria	1stF	2ndF	3rdF	4thF	5thF	6thF	7thF	8thF	9thF	10thF	AVG
Output from All	Mean DSC	0.84	0.61	0.90	0.85	0.83	0.91	0.87	0.87	0.87	0.87	0.84
	Median DSC	0.83	0.62	0.91	0.86	0.86	0.91	0.91	0.89	0.87	0.88	0.85
	STD-dev	0.03	0.1	0.03	0.06	0.07	0.01	0.08	0.05	0.03	0.04	0.05
Non-bypass	Mean DSC	0.83	0.81	0.89	0.71	0.83	0.89	0.87	0.87	0.87	0.83	0.84
	Median DSC	0.86	0.82	0.90	0.87	0.86	0.90	0.86	0.88	0.88	0.86	0.87
	STD-dev	0.1	0.04	0.02	0.2	0.07	0.01	0.03	0.04	0.02	0.04	0.05

TABLE A.6: Performance of the Output from All and Non-bypass models with using various number of layers (4-4-5-7-10-12-15). M, Model; F, Fold

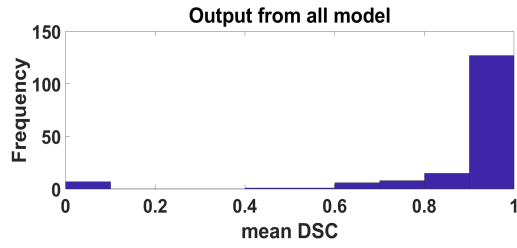




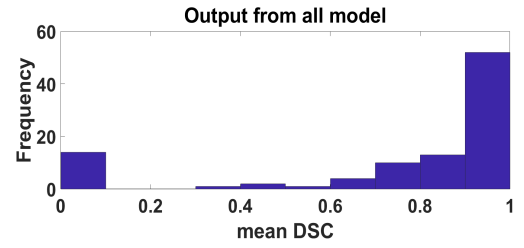
(a) 1st fold



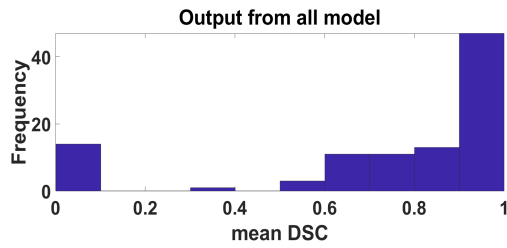
(b) 2nd fold



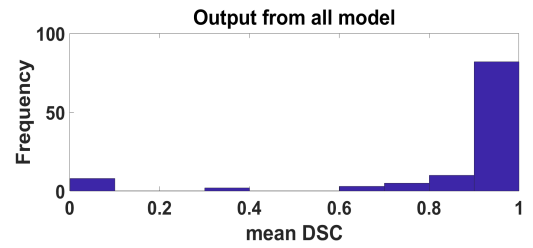
(c) 3rd fold



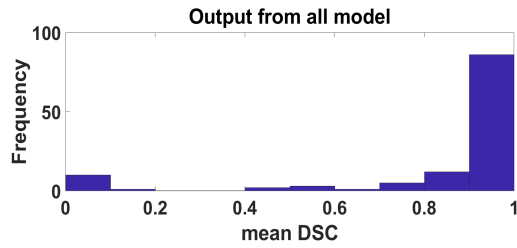
(d) 4th fold



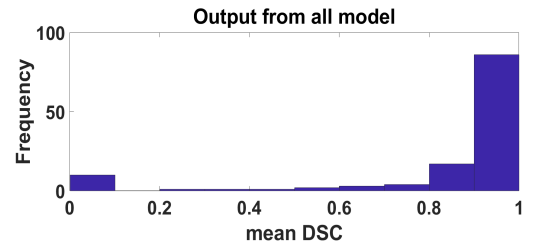
(e) 5th fold



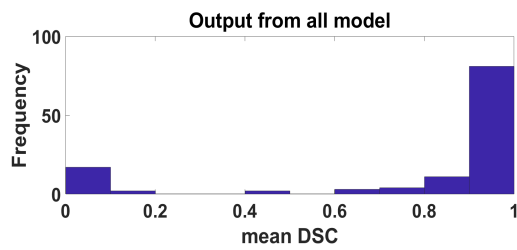
(f) 6th fold



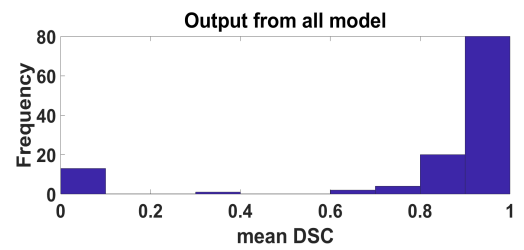
(g) 7th fold



(h) 8th fold

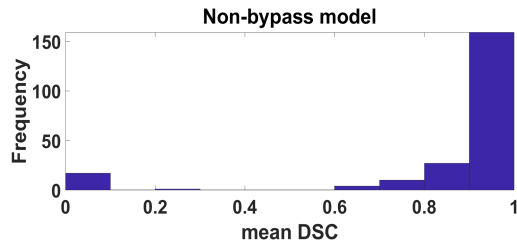


(i) 9th fold

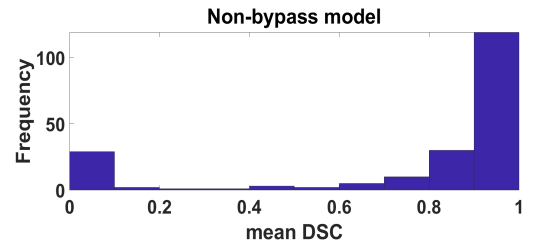


(j) 10th fold

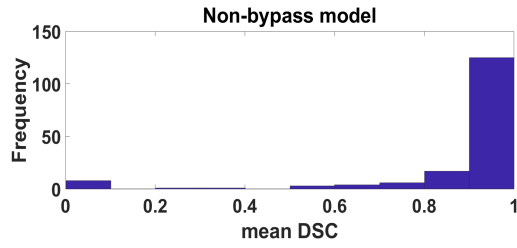
FIGURE A.2: Obtained results of the Output from All model on the all ten folds (test images).



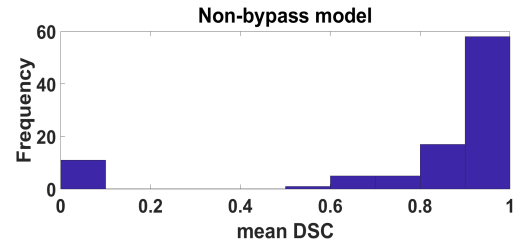
(a) 1st fold



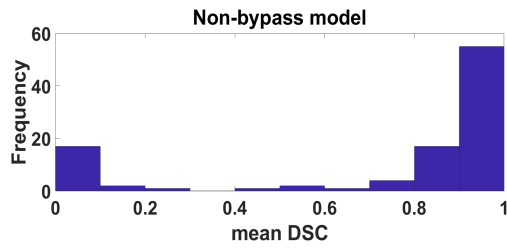
(b) 2nd fold



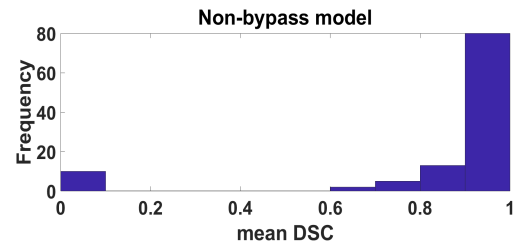
(c) 3rd fold



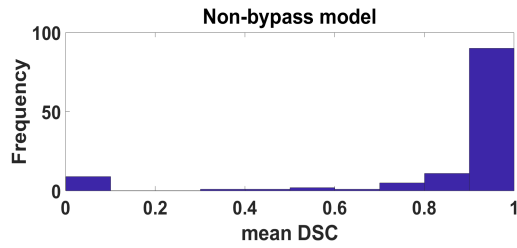
(d) 4th fold



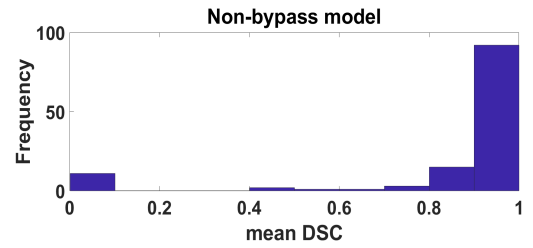
(e) 5th fold



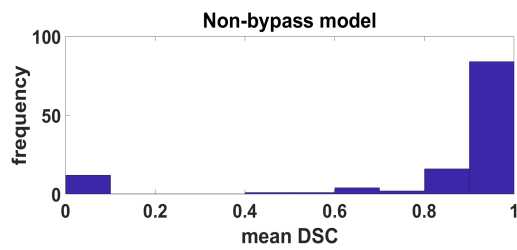
(f) 6th fold



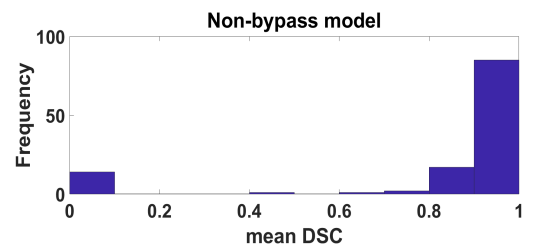
(g) 7th fold



(h) 8th fold



(i) 9th fold



(j) 10th fold

FIGURE A.3: Obtained results of the Non-bypass model on the all ten folds (test image).

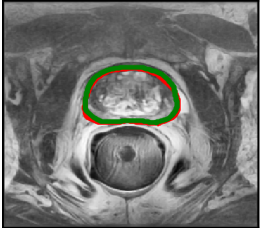
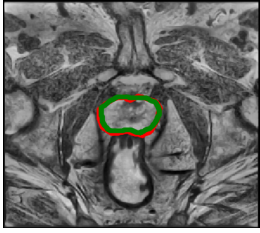
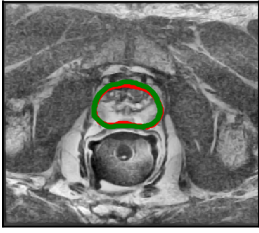
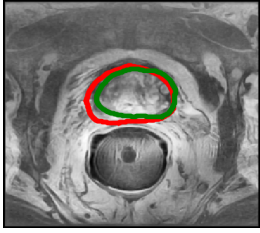
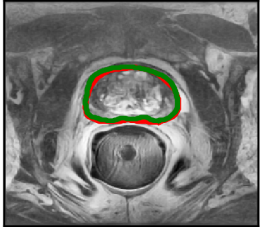
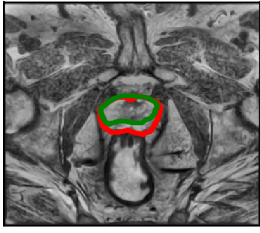
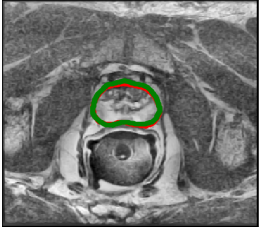
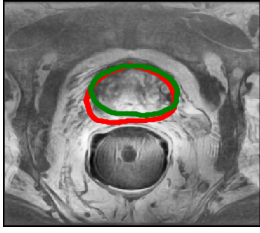
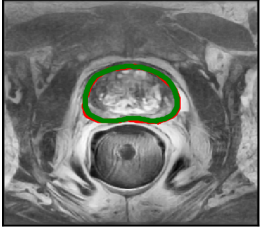

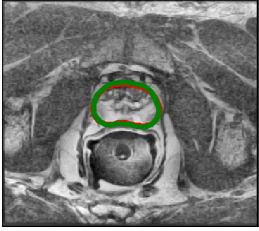
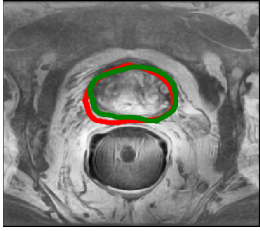
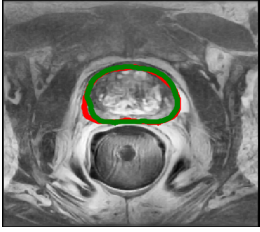
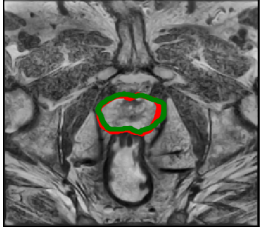
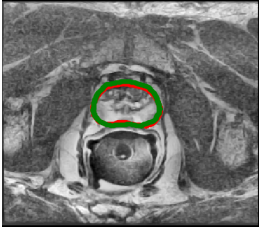
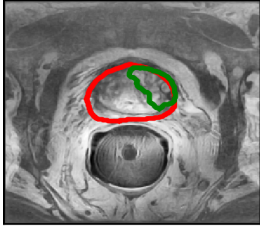
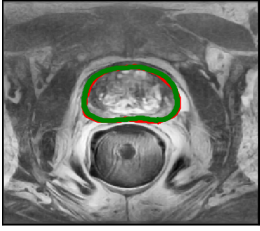
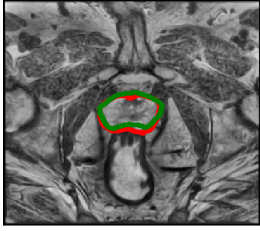
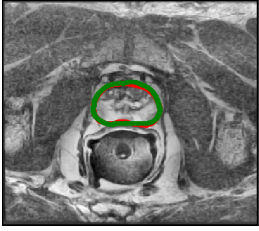
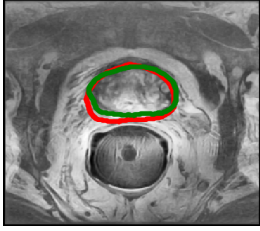
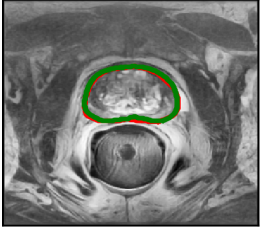
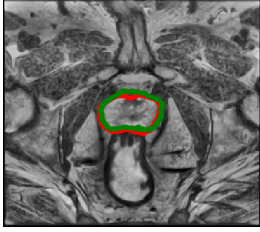
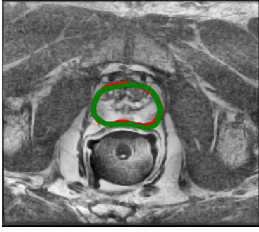
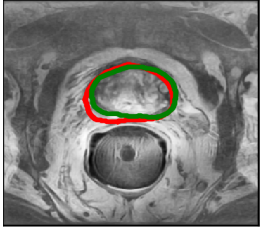
Model	a	b	c	d
Straight(Baseline)				
Bypass				
Output from All				
Input to All				
Dense				
Non-bypass				

TABLE A.7: Segmentation of four different images from the test set of the first fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block.



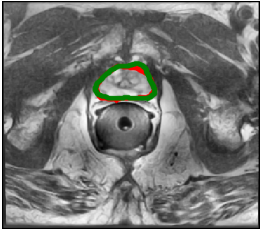
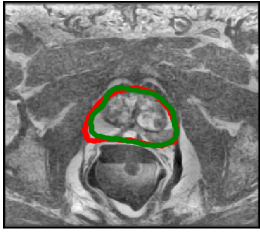
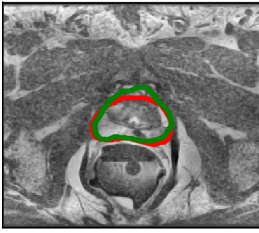
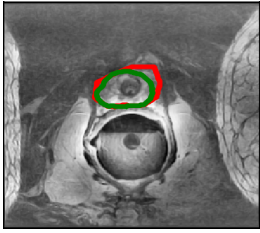
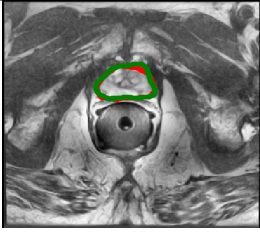


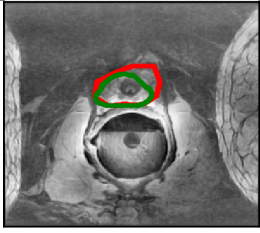
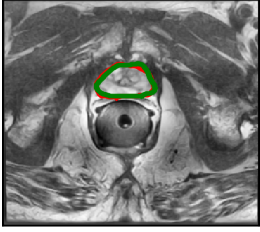


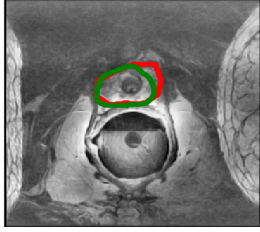
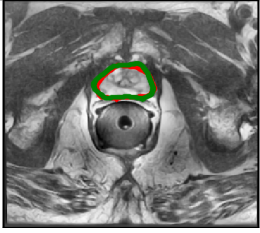


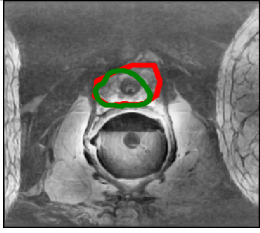
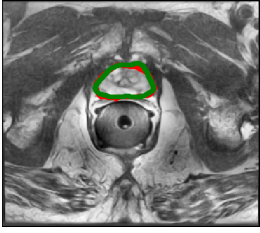
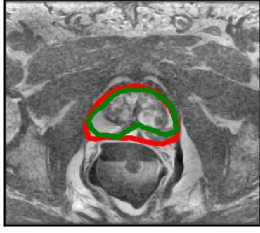

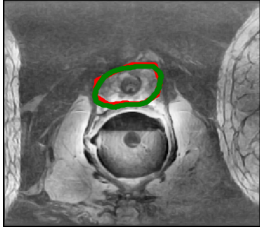
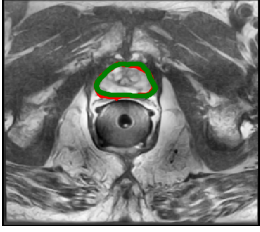

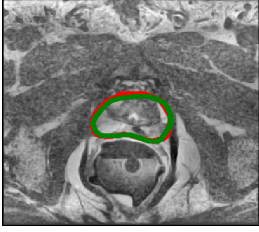
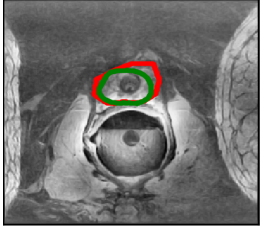
Model	a	b	c	d
Straight (Baseline)				
Bypass				
Output from All				
Input to All				
Dense				
Non-bypass				

TABLE A.8: Segmentation of four different images from the test set of the second fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block.

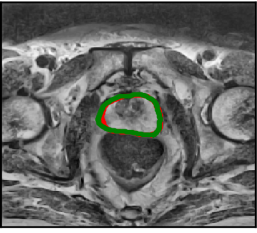
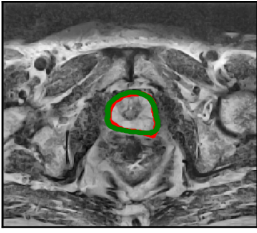
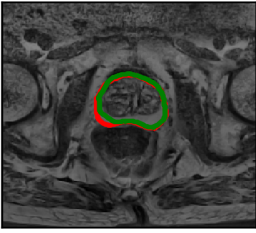
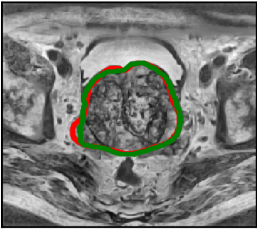
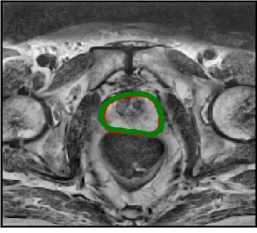
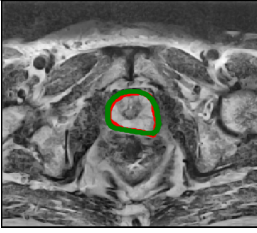
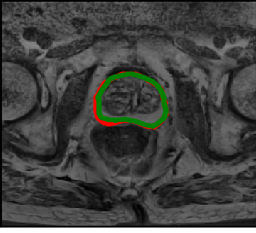
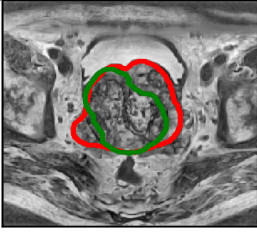

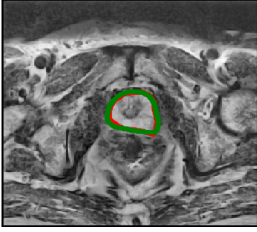
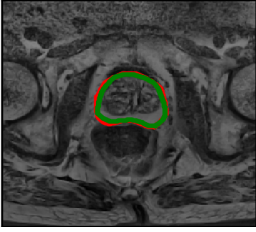
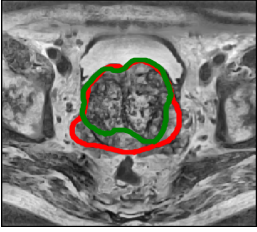

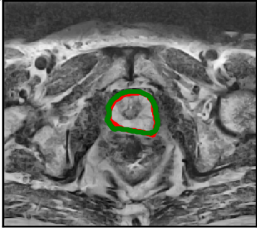
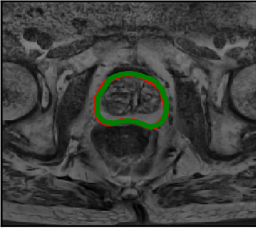
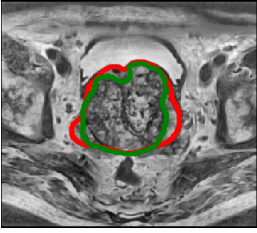

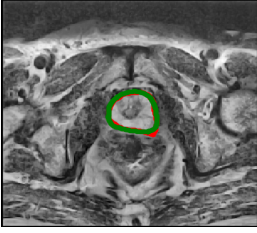
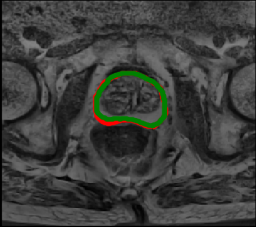
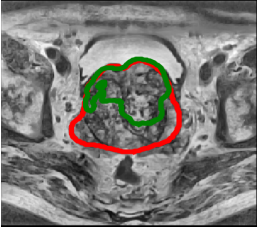


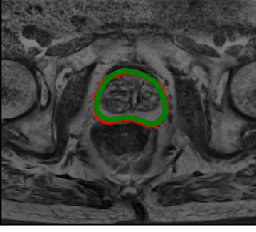
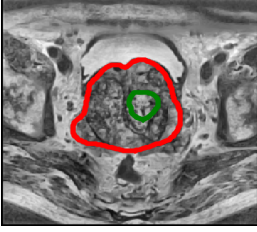
Model	a	b	c	d
Straight (Baseline)				
Bypass				
Output from All				
Input to All				
Dense				
Non-bypass				

TABLE A.9: Segmentation of four different images from the test set of the fifth fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block.



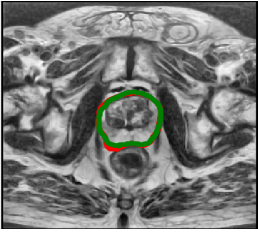
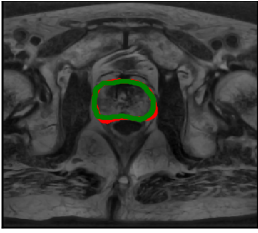
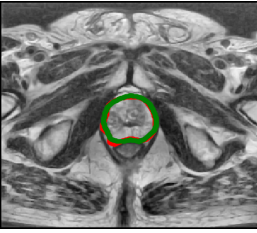
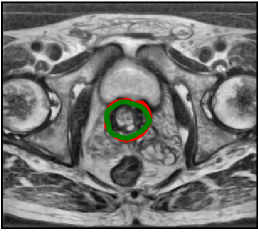
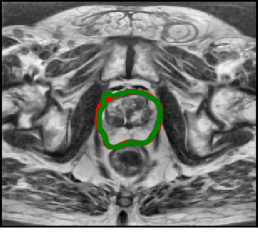
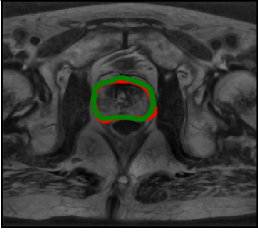
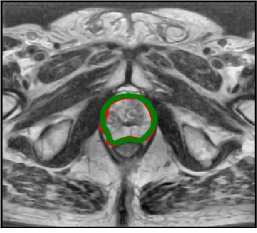
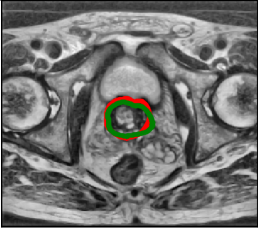

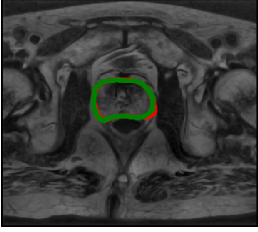
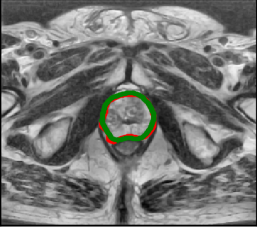
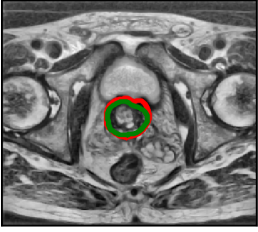
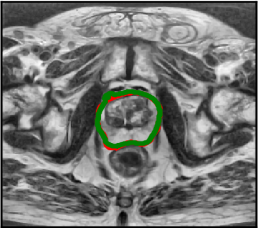
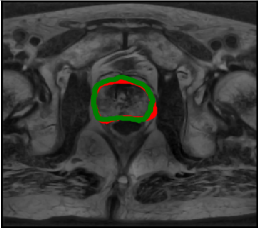
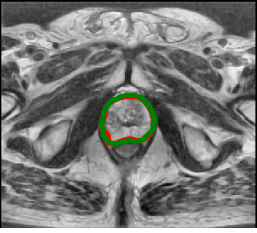

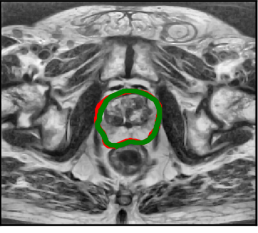
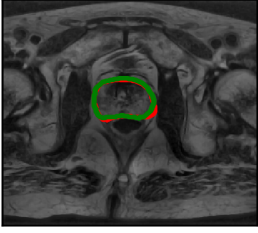
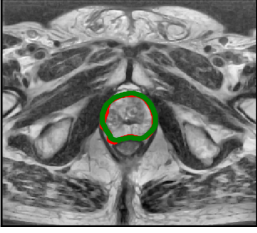

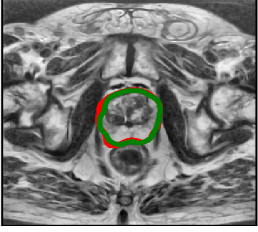
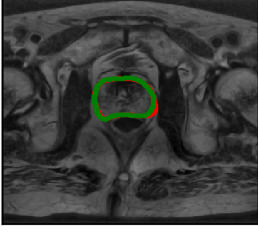
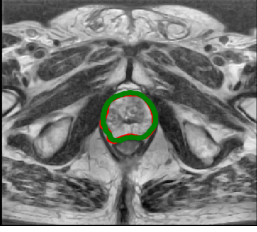
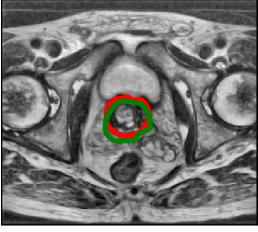
Model	a	b	c	d
Straight (Baseline)				
Bypass				
Output from All				
Input to All				
Dense				
Non-bypass				

TABLE A.10: Segmentation of four different images from the test set of the sixth fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block.

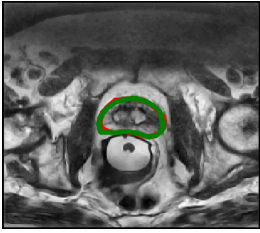
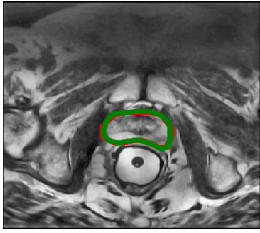
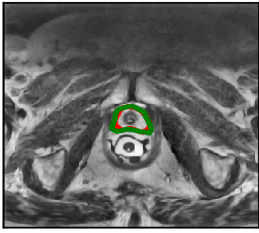
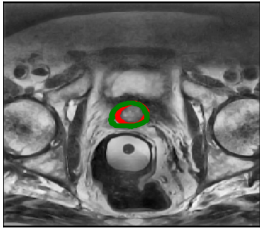
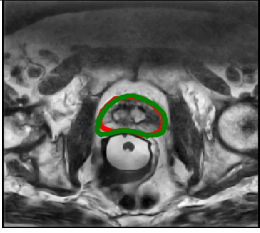
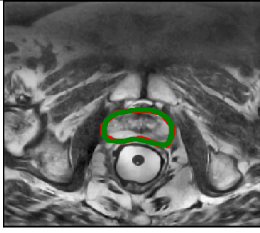
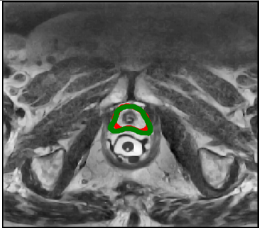
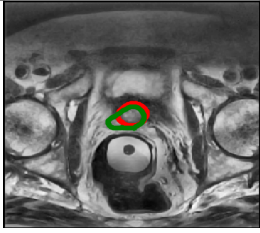
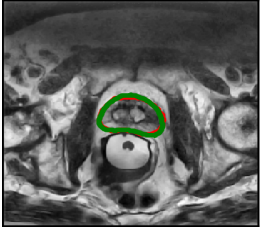
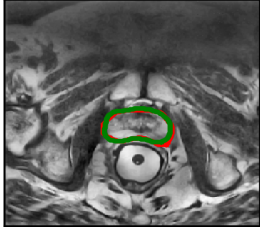
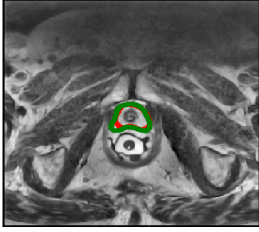
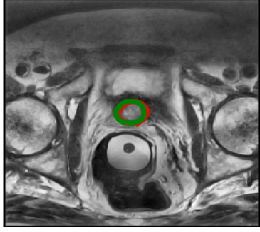
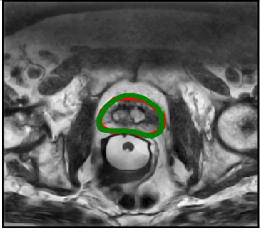
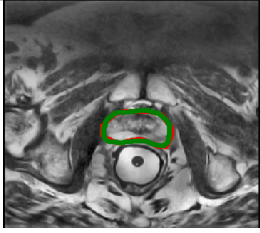
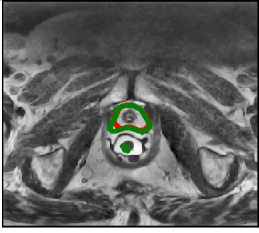
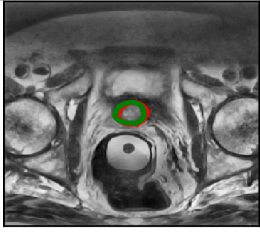
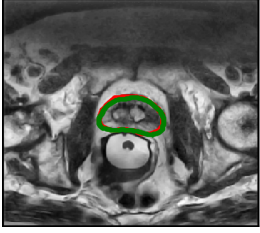
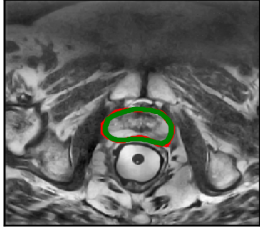
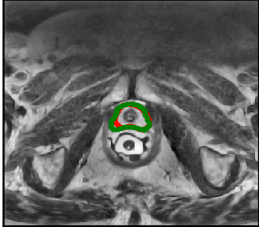
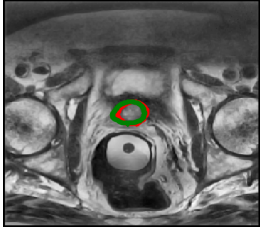
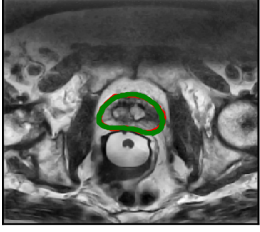
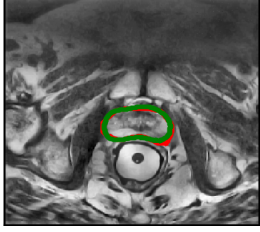
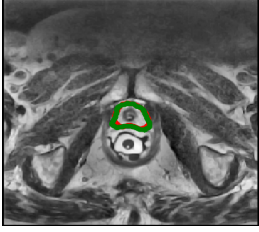
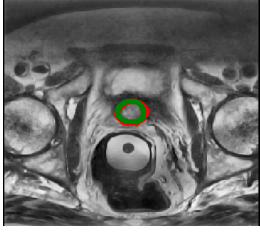
Model	a	b	c	d
Straight (Baseline)				
Bypass				
Output from All				
Input to All				
Dense				
Non-bypass				

TABLE A.11: Segmentation of four different images from the test set of the tenth fold using six proposed networks. The red border is the ground truth, and green is predicted border. Straight, the Straight model with three layers per block; Bypass, the Bypass model with three layers per block; Output from All, the Output from All model with seven layers per block; Input to All, the Input to All model with two layers per block; Dense, the Dense model with three layers per block; Non-bypass, the Non-bypass model with seven layers per block.





# List of Symbols

MRI Magnetic Resonance Image

CNN Convolutional Neural Network

FCNN Fully Convolutional Neural Network

ROI Region Of Interest

2D 2 Dimensional

3D 3 Dimensional

IBM International Business Machines corporation

DSC Dice Similarity Coefficient

SSAM Statistical Shape and Appearance Model

AAM Active Appearance Model

SVM Support Vector Machine

VOI Volume Of Interest

MAP Maximum A Posteriori

ASM Active Shape Model

ACM Active Contour Model

SLIC	Simple Linear Iterative Clustering method
HNN	Holistically Nested Network
CED	Coherence Enhanced Diffusion
AFL-PBT	Adaptive Feature Learning Probability Boosting Tree
GOP	Geodesic Object Proposal
SSAE	Stacked Sparse Auto Encoder
ISA	Independent Subspace Analysis
BCNN	Baseline Convolutional Neural Network
ERC	EndoRectal Coil
non-ERC	non- EndoRectal Coil
BN	Batch Normalization
ReLU	Rectified Linear Unit
$B$	Mini-batch
$m$	Size of the batch
$\gamma$	Scale parameter
$\beta$	Shift parameter
$\sigma$	Variance
SGD	Stochastic Gradient Descent
ADAM	Adaptive Moment Estimation
ADAM1	ADAM with learning rate 0.01
ADAM2	ADAM with learning rate 0.001

ADAM3 ADAM with learning rate 0.0001

SGD Stochastic Gradient Descent

SGD SGD with learning rate 0.01

SGD SGD with learning rate 0.001

SGD SGD with learning rate 0.0001

Droupout1 Dropout after each layer with probability of 0.2

Droupout2 Dropout end of the block with probability of 0.2

Droupout3 Dropout in the bottleneck with probability of 0.5

S2 Straight model with two layers

S3 Straight model with three layers

B2 Bypass model with two layers

B3 Bypass model with three layers

I2 Input to all with two layers

I3 Input to all with three layers

I5 Input to all with five layers

O2 Output from all with using two layers

O3 Output from all with using three layers

O7 Output from all with using seven layers

F2 Dense with using two layers

F3 Dense with using three layers

NP2 Non-bypass with using two layers

NP3 Non-bypass with using three layers

NP7 Non-bypass with using seven layers

NSep Normalise Separately

NAll Normalise All

NPix Normalise Pixels

HUH Haukeland University Hospital

BIDMC Beth Israel Deaconess Medical Center

UCL University College London

RUNMC Radboud University Nijmegen Medical Center

## References

- [1] *The digital universe driving data growth in healthcare-* published by EMC with research and analysis from IDC (12/13). <https://www.emc.com/analyst-report/digital-universe-healthcare-vertical-report-ar.pdf>. 1
- [2] [https://www-03.ibm.com/press/us/en/pressrelease/51146.wss#\\_ftnref1](https://www-03.ibm.com/press/us/en/pressrelease/51146.wss#_ftnref1). 1
- [3] K. H. Leissner and L. E. Tisell. *The weight of the human prostate*. Scandinavian Journal of Urology and Nephrology **13**(2), 137 (1979). 1
- [4] G. P. Haas, N. Delongchamps, O. W. Brawley, C. Y. Wang, and G. de la Roza. *The worldwide epidemiology of prostate cancer: perspectives from autopsy studies*. The Canadian Journal of Urology **15**(1), 3866 (2008). 2
- [5] R. Siegel, M. Kimberly, and A. Jemal. *Cancer statistics, 2018*. CA: A Cancer Journal for Clinicians **68**, 7 (2018). 2
- [6] *Prostate cancer foundation of australia*. <http://www.prostate.org.au/awareness/general-information/what-you-need-to-know-about-prostate-cancer/>. 2
- [7] *Australia institute of health and welfare 2017*. Australian Cancer Incidence and Mortality (ACIM) books: Prostate cancer <http://www.aihw.gov.au/acim-books>. 2
- [8] *Australia institute of health and welfare 2017*. *cancer in Australia 2017*. Cancer Series **101**, Cat. No. CAN 100 (2017). 2

- [9] <https://www.seattlecca.org/diseases/prostate-cancer/early-detection-prevention>. 2
- [10] N. Makni, P. Puech, R. Lopes, A. S. Dewalle, O. Colot, and N. Betrouni. *Combining a deformable model and a probabilistic framework for an automatic 3D segmentation of prostate on MRI*. International Journal of Computer Assisted Radiology and Surgery **4**(2), 181 (2009). 2
- [11] S. Martin, V. Daanen, and J. Troccaz. *Atlas-based prostate segmentation using an hybrid registration*. International Journal of Computer Assisted Radiology and Surgery **3**(6), 485 (2008). 2
- [12] B. Turkbey and P. L. Choyke. *Multiparametric MRI and prostate cancer diagnosis and risk stratification*. Current Opinion in Urology **22**(4), 310 (2012). 2
- [13] X. Huang and G. Tsechpenakis. *Medical image segmentation*. Information Discovery on Electronic Health Records **10**, 251 (2009). 3
- [14] N. Sharma and L. M. Aggarwal. *Automated medical image segmentation techniques*. Journal of Medical Physics/Association of Medical Physicists of India **35**(1), 3 (2010). 3
- [15] M. S. Fasihi and W. B. Mikhael. *Overview of current biomedical image segmentation methods*. In *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 803–808 (IEEE, 2016). 3
- [16] D. R. White, A. S. Houston, W. F. Sampson, and G. P. Wilkins. *Intra-and interoperator variations in region-of-interest drawing and their effect on the measurement of glomerular filtration rates*. Clinical Nuclear Medicine **24**(3), 177 (1999). 3
- [17] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, vol. 1 (MIT press Cambridge, 2016). 4
- [18] L. Deng, D. Yu, et al. *Deep learning: methods and applications*. Foundations and Trends<sup>®</sup> in Signal Processing **7**(3–4), 197 (2014). 4

- [19] H. C. Shin, K. Roberts, L. Lu, D. Demner Fushman, J. Yao, and R. M. Summers. *Learning to read chest X-rays: Recurrent neural cascade model for automated image annotation*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2497–2506 (2016). [4](#), [12](#)
- [20] H. I. Suk, S. W. Lee, D. Shen, A. D. N. Initiative, et al. *Latent feature representation with stacked auto-encoder for AD/MCI diagnosis*. *Brain Structure and Function* **220**(2), 841 (2015). [4](#), [12](#)
- [21] J. Long, E. Shelhamer, and T. Darrell. *Fully convolutional networks for semantic segmentation*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440 (2015). [4](#), [14](#)
- [22] O. Ronneberger, P. Fischer, and T. Brox. *U-net: Convolutional networks for biomedical image segmentation*. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241 (Springer, 2015). [4](#), [14](#), [20](#), [22](#)
- [23] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. *Densely connected convolutional networks*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 3 (2017). [4](#), [29](#)
- [24] K. He, X. Zhang, S. Ren, and J. Sun. *Deep residual learning for image recognition*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016). [4](#), [22](#), [23](#), [27](#)
- [25] Y. LeCun, Y. Bengio, and G. Hinton. *Deep learning*. *Nature* **521**(7553), 436 (2015). [7](#)
- [26] G. Litjens, R. Toth, W. van de Ven, C. Hoeks, S. Kerkstra, B. van Ginneken, G. Vincent, G. Guillard, N. Birbeck, J. Zhang, et al. *Evaluation of prostate segmentation algorithms for MRI: the PROMISE12 challenge*. *Medical Image Analysis* **18**(2), 359 (2014). [33](#)
- [27] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sánchez. *A survey on deep learning in medical image analysis*. *Medical Image Analysis* **42**, 60 (2017).

- [28] D. Shen, G. Wu, and H. I. Suk. *Deep learning in medical image analysis*. Annual Review of Biomedical Engineering **19**, 221 (2017). [7](#)
- [29] S. Klein, U. A. Van Der Heide, I. M. Lips, M. Van Vulpen, M. Staring, and J. P. Pluim. *Automatic segmentation of the prostate in 3D MR images by atlas matching using localized mutual information*. Medical Physics **35**(4), 1407 (2008). [7](#), [36](#)
- [30] T. Heimann, B. Van Ginneken, M. A. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes, *et al.* *Comparison and evaluation of methods for liver segmentation from CT datasets*. IEEE Transactions on Medical Imaging **28**(8), 1251 (2009).
- [31] T. Sorensen. *A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons*. Biol. Skr. **5**, 1 (1948).
- [32] L. R. Dice. *Measures of the amount of ecologic association between species*. Ecology **26**(3), 297 (1945). [7](#), [36](#)
- [33] B. Zitova and J. Flusser. *Image registration methods: a survey*. Image and Vision Computing **21**(11), 977 (2003). [7](#)
- [34] J. C. Gee, M. Reivich, and R. Bajcsy. *Elastically deforming a three-dimensional atlas to match anatomical brain images* (1993). [8](#)
- [35] M. R. Sabuncu, B. T. Yeo, K. Van Leemput, B. Fischl, and P. Golland. *A generative model for image segmentation based on label fusion*. IEEE Transactions on Medical Imaging **29**(10), 1714 (2010). [8](#)
- [36] J. P. Thirion. *Image matching as a diffusion process: an analogy with maxwell's demons*. Medical Image Analysis **2**(3), 243 (1998). [8](#)
- [37] S. Ghose, J. Mitra, A. Oliver, R. Marti, X. Llado, J. Freixenet, J. C. Vilanova, D. Sidibé, and F. Mériaudeau. *Graph cut energy minimization in a probabilistic learning framework for 3D prostate segmentation in MRI*. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pp. 125–128 (IEEE, 2012). [8](#), [9](#), [17](#)



- [38] Q. Gao, A. Asthana, T. Tong, Y. Hu, D. Rueckert, and P. Edwards. *Hybrid decision forests for prostate segmentation in multi-channel MR images*. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pp. 3298–3303 (IEEE, 2014). [8](#)
- [39] S. Ghose, J. Mitra, A. Oliver, R. Marti, X. Llado, J. Freixenet, J. C. Vilanova, D. Sidibe, and F. Meriaudeau. *A coupled schema of probabilistic atlas and statistical shape and appearance model for 3D prostate segmentation in MR images*. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pp. 541–544 (IEEE, 2012). [8](#)
- [40] A. Li, C. Li, X. Wang, S. Eberl, D. D. Feng, and M. Fulham. *Automated segmentation of prostate MR images using prior knowledge enhanced random walker*. In *Digital Image Computing: Techniques and Applications (DICTA), 2013 International Conference on*, pp. 1–7 (IEEE, 2013). [8](#)
- [41] S. Martin, J. Troccaz, and V. Daanen. *Automated segmentation of the prostate in 3D MR images using a probabilistic atlas and a spatially constrained deformable model*. *Medical Physics* **37**(4), 1579 (2010). [8](#)
- [42] R. Cheng, B. Turkbey, W. Gandler, H. K. Agarwal, V. P. Shah, A. Bokinsky, E. McCreedy, S. Wang, S. Sankineni, M. Bernardo, *et al.* *Atlas based AAM and SVM model for fully automatic MRI prostate segmentation*. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE*, pp. 2881–2585 (IEEE, 2014). [8](#)
- [43] A. S. Korsager, V. Fortunati, F. Lijn, J. Carl, W. Niessen, L. R. Ostergaard, and T. Walsum. *The use of atlas registration and graph cuts for prostate segmentation in magnetic resonance images*. *Medical Physics* **42**(4), 1614 (2015). [8](#)
- [44] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. *Active shape models-their training and application*. *Computer Vision and Image Understanding* **61**(1), 38 (1995). [9](#)
- [45] T. F. Cootes, G. J. Edwards, and C. J. Taylor. *Active appearance models*. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(6), 681 (2001). [9](#)
- [46] A. Firjani, F. Khalifa, A. Elnakib, G. Gimel'Farb, M. A. El-Ghar, A. Elmaghraby, and A. El-Baz. *3D automatic approach for precise segmentation of the prostate from diffusion-weighted*

- magnetic resonance imaging. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pp. 2285–2288 (IEEE, 2011). [9](#)
- [47] A. Firjani, A. Elnakib, F. Khalifa, G. Gimel'Farb, M. A. El-Ghar, J. Suri, A. Elmaghraby, and A. El-Baz. *A new 3D automatic segmentation framework for accurate segmentation of prostate from DCE-MRI*. In *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*, pp. 1476–1479 (IEEE, 2011). [9](#), [10](#), [17](#)
- [48] A. Firjani, A. Elnakib, F. Khalifa, A. El-Baz, G. Gimel'Farb, M. A. El-Ghar, and A. Elmaghraby. *A novel 3D segmentation approach for segmenting the prostate from dynamic contrast enhanced MRI using current appearance and learned shape prior*. In *Signal Processing and Information Technology (ISSPIT), 2010 IEEE International Symposium on*, pp. 137–143 (IEEE, 2010). [9](#)
- [49] R. Toth and A. Madabhushi. *Multifeature landmark-free active appearance models: application to prostate MRI segmentation*. *IEEE Transactions on Medical Imaging* **31**(8), 1638 (2012). [9](#), [10](#)
- [50] R. Toth, J. Ribault, J. Gentile, D. Sperling, and A. Madabhushi. *Simultaneous segmentation of prostatic zones using active appearance models with multiple coupled levelsets*. *Computer Vision and Image Understanding* **117**(9), 1051 (2013). [9](#), [10](#)
- [51] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. *Fast geodesic active contours*. *IEEE Transactions on Image Processing* **10**(10), 1467 (2001). [10](#)
- [52] A. Skalski, J. Lagwa, P. Kedzierawski, T. Zielinski, and T. Kuszewski. *Automatic prostate segmentation in MR images based on 3D active contours with shape constraints*. In *Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), 2013*, pp. 246–249 (IEEE, 2013). [10](#)
- [53] J. H. Liew, W. Xiong, Y. Gu, J. Cheng, and S. H. Ong. *3D prostate segmentation from MRI images using modified rotational slice-based level set with non-uniform contour shrinking*. In *Industrial Electronics and Applications (ICIEA), 2015 IEEE 10th Conference on*, pp. 224–228 (IEEE, 2015). [10](#)

- [54] X. Yang, S. Zhan, and D. Xie. *Landmark based prostate MRI segmentation via improved level set method*. In *Signal Processing (ICSP), 2016 IEEE 13th International Conference on*, pp. 29–34 (IEEE, 2016). [10](#), [17](#)
- [55] X. Wang, W. Tang, H. Cui, S. Zeng, D. D. Feng, and M. Fulham. *Multi-view collaborative segmentation for prostate MRI images*. In *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE*, pp. 3529–3532 (IEEE, 2017). [10](#), [11](#)
- [56] B. Ajani and K. Krishnan. *Automatic and interactive prostate segmentation in MRI using learned contexts on a sparse graph template*. In *Biomedical Imaging (ISBI 2017), 2017 IEEE 14th International Symposium on*, pp. 315–318 (IEEE, 2017). [10](#), [11](#)
- [57] K. Wu, C. Garnier, H. Shu, and J. L. Dillenseger. *Prostate segmentation on T2 MRI using optimal surface detection*. *Innovation and Research in BioMedical engineering* **34**(4-5), 287 (2013). [10](#), [11](#)
- [58] X. Ren and J. Malik. *Learning a classification model for segmentation*. In *Ninth IEEE International Conference on Computer Vision*, p. 10 (IEEE, 2003). [11](#)
- [59] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Ssstrunk. *SLIC superpixels compared to state-of-the-art superpixel methods*. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(11), 2274 (2012). [11](#)
- [60] S. Lloyd. *Least squares quantization in pcm*. *IEEE transactions on information theory* **28**(2), 129 (1982). [11](#)
- [61] Z. Tian, L. Liu, Z. Zhang, and B. Fei. *Superpixel-based segmentation for 3D prostate MR images*. *IEEE Transactions on Medical Imaging* **35**(3), 791 (2016). [11](#), [12](#)
- [62] D. Mahapatra and J. M. Buhmann. *Prostate MRI segmentation using learned semantic knowledge and graph cuts*. *IEEE Transactions on Biomedical Engineering* **61**(3), 756 (2014). [11](#), [12](#)
- [63] D. Mahapatra. *Graph cut based automatic prostate segmentation using learned semantic information*. In *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pp. 1316–1319 (IEEE, 2013). [11](#), [12](#), [17](#)

- [64] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio. *Object recognition with gradient-based learning*. In *Shape, contour and grouping in computer vision*, pp. 319–345 (Springer, 1999). [12](#)
- [65] M. D. Zeiler and R. Fergus. *Visualizing and understanding convolutional networks*. In *European conference on computer vision*, pp. 818–833 (Springer, 2014). [12](#), [21](#)
- [66] R. Cheng, H. R. Roth, L. Lu, S. Wang, B. Turkbey, W. Gandler, E. S. McCreedy, H. K. Agarwal, P. Choyke, R. M. Summers, et al. *Active appearance model and deep learning for more accurate prostate segmentation on MRI*. In *Medical Imaging 2016: Image Processing*, vol. 9784, p. 97842I (International Society for Optics and Photonics, 2016). [12](#), [16](#), [17](#)
- [67] A. Krizhevsky, I. Sutskever, and G. E. Hinton. *Imagenet classification with deep convolutional neural networks*. In *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012). [12](#)
- [68] R. Cheng, H. R. Roth, N. S. Lay, L. Lu, B. Turkbey, W. Gandler, E. S. McCreedy, T. J. Pohida, P. A. Pinto, P. L. Choyke, et al. *Automatic magnetic resonance prostate segmentation by deep learning with holistically nested networks*. *Journal of Medical Imaging* **4**(4), 041302 (2017). [12](#), [16](#)
- [69] H. Jia, Y. Xia, Y. Song, W. Cai, M. Fulham, and D. D. Feng. *Atlas registration and ensemble deep convolutional neural network-based prostate segmentation using magnetic resonance imaging*. *Neurocomputing* **275**, 1358 (2018). [13](#), [16](#), [17](#)
- [70] K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556 (2014). [13](#), [20](#), [27](#)
- [71] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. *Gradient-based learning applied to document recognition*. *Proceedings of the IEEE* **86**(11), 2278 (1998). [13](#)
- [72] B. He, D. Xiao, Q. Hu, and F. Jia. *Automatic magnetic resonance image prostate segmentation based on adaptive feature learning probability boosting tree initialization and CNN-ASM refinement*. *IEEE Access* **6**, 2005 (2018). [13](#), [16](#)

- [73] K. Yan, C. Li, X. Wang, A. Li, Y. Yuan, D. Feng, M. Khadra, and J. Kim. *Automatic prostate segmentation on MR images with deep network and graph model*. In *Engineering in Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the*, pp. 635–638 (IEEE, 2016). [13](#), [16](#)
- [74] P. Krähenbühl and V. Koltun. *Geodesic object proposals*. In *European Conference on Computer Vision*, pp. 725–739 (Springer, 2014). [13](#)
- [75] Y. Guo, Y. Gao, and D. Shen. *Deformable MR prostate segmentation via deep feature learning and sparse patch matching*. In *Deep Learning for Medical Image Analysis*, pp. 197–222 (Elsevier, 2017). [13](#), [16](#)
- [76] S. Liao, Y. Gao, A. Oto, and D. Shen. *Representation learning: a unified deep learning framework for automatic prostate MR segmentation*. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 254–261 (Springer, 2013). [13](#), [16](#)
- [77] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. *Deconvolutional networks* (2010). [14](#), [21](#)
- [78] D. Cireşan, U. Meier, and J. Schmidhuber. *Multi-column deep neural networks for image classification*. arXiv preprint arXiv:1202.2745 (2012). [14](#), [21](#)
- [79] Q. Zhu, B. Du, B. Turkbey, P. L. Choyke, and P. Yan. *Deeply-supervised CNN for prostate segmentation*. In *Neural Networks (IJCNN), 2017 International Joint Conference on*, pp. 178–184 (IEEE, 2017). [14](#), [16](#), [36](#), [40](#), [52](#)
- [80] F. Milletari, N. Navab, and S.-A. Ahmadi. *V-net: Fully convolutional neural networks for volumetric medical image segmentation*. In *3D Vision (3DV), 2016 Fourth International Conference on*, pp. 565–571 (IEEE, 2016). [15](#), [16](#), [52](#)
- [81] L. Yu, X. Yang, H. Chen, J. Qin, and P.-A. Heng. *Volumetric ConvNets with mixed residual connections for automated prostate segmentation from 3D MR images*. In *Association for the Advancement of Artificial Intelligence*, pp. 66–72 (2017). [15](#), [16](#), [20](#), [22](#), [40](#), [48](#), [52](#)

- [82] J. Mun, W.-D. Jang, D. J. Sung, and C.-S. Kim. *Comparison of objective functions in CNN-based prostate magnetic resonance image segmentation*. In *Image Processing (ICIP), 2017 IEEE International Conference on*, pp. 3859–3863 (IEEE, 2017). 16, 36, 40, 48, 52
- [83] *SPIE-AAPM-NCI prostatex challenges*. <https://wiki.cancerimagingarchive.net/display/Public/SPIE-AAPM-NCI+PROSTATEx+Challenges>. 16
- [84] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. *Going deeper with convolutions*. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9 (2015). 20
- [85] T. Dietterich. *Overfitting and undercomputing in machine learning*. *ACM computing surveys (CSUR)* 27(3), 326 (1995). 20
- [86] M. Drozdal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal. *The importance of skip connections in biomedical image segmentation*. In *Deep Learning and Data Labeling for Medical Applications*, pp. 179–187 (Springer, 2016). 20
- [87] R. K. Srivastava, K. Greff, and J. Schmidhuber. *Highway networks*. arXiv preprint arXiv:1505.00387 (2015). 22
- [88] R. H. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung. *Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit*. *Nature* 405(6789), 947 (2000). 23, 27
- [89] S. Ioffe and C. Szegedy. *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. arXiv preprint arXiv:1502.03167 (2015). 23, 25, 26
- [90] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. *Improving neural networks by preventing co-adaptation of feature detectors*. arXiv preprint arXiv:1207.0580 (2012). 23, 25
- [91] H. Shimodaira. *Improving predictive inference under covariate shift by weighting the log-likelihood function*. *Journal of Statistical Planning and Inference* 90(2), 227 (2000). 26

- [92] *Deeplearning.ai: Why does batch norm work? (c2w3l06)*. <https://www.youtube.com/watch?v=nUUqwaxLnWs>. 26
- [93] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. *Dropout: a simple way to prevent neural networks from overfitting*. *The Journal of Machine Learning Research* **15**(1), 1929 (2014). 26
- [94] X. Glorot, A. Bordes, and Y. Bengio. *Deep sparse rectifier neural networks*. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 315–323 (2011). 27
- [95] Y. Bengio, P. Simard, and P. Frasconi. *Learning long-term dependencies with gradient descent is difficult*. *IEEE Transactions on Neural Networks* **5**(2), 157 (1994). 27
- [96] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. *Learning internal representations by error propagation*. Tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science (1985). 27
- [97] Y. Zhang, T. Shen, X. Ji, Y. Zhang, R. Xiong, and Q. Dai. *Residual highway convolutional neural networks for in-loop filtering in hevc*. *IEEE Transactions on Image Processing* **27**(8), 3827 (2018). 27
- [98] S. Jégou, M. Drozdal, D. Vazquez, A. Romero, and Y. Bengio. *The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation*. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pp. 1175–1183 (IEEE, 2017). 29
- [99] <https://keras.io/>. 34
- [100] <https://keras.io/preprocessing/image/>. 34
- [101] P. Y. Simard, D. Steinkraus, J. C. Platt, et al. *Best practices for convolutional neural networks applied to visual document analysis*. In *International Conference on Document Analysis and Recognition*, vol. 3, pp. 958–962 (2003). 34
- [102] E. Kreyszig. *Advanced engineering mathematics* (John Wiley & Sons, 2010). 35



- [103] H. Robbins and S. Monro. *A stochastic approximation method*. In *Herbert Robbins Selected Papers*, pp. 102–109 (Springer, 1985). [37](#)
- [104] D. P. Kingma and J. Ba. *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980 (2014). [37](#)
- [105] F. Wilcoxon. *Individual comparisons by ranking methods*. Biometrics bulletin **1**(6), 80 (1945). [49](#)
- [106] D. Ji, J. Yu, T. Kurihara, L. Xu, and S. Zhan. *Automatic prostate segmentation on mr images with deeply supervised network*. In *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 309–314 (IEEE, 2018). [52](#)
- [107] T. Clark, J. Zhang, S. Baig, A. Wong, M. A. Haider, and F. Khalvati. *Fully automated segmentation of prostate whole gland and transition zone in diffusion-weighted mri using convolutional neural networks*. Journal of Medical Imaging **4**(4), 041307 (2017). [52](#)
- [108] W. Chen, Y. Zhang, J. He, Y. Qiao, Y. Chen, H. Shi, and X. Tang. *W-net: Bridged u-net for 2d medical image segmentation*. arXiv preprint arXiv:1807.04459 (2018). [52](#)
- [109] Q. Liu, M. Fu, X. Gong, and H. Jiang. *Densely dilated spatial pooling convolutional network using benign loss functions for imbalanced volumetric prostate segmentation*. arXiv preprint arXiv:1801.10517 (2018). [52](#)
- [110] M. Drozdal, G. Chartrand, E. Vorontsov, M. Shakeri, L. Di Jorio, A. Tang, A. Romero, Y. Bengio, C. Pal, and S. Kadoury. *Learning normalized inputs for iterative estimation in medical image segmentation*. Medical image analysis **44**, 1 (2018). [52](#)
- [111] J. Sun, Y. Shi, Y. Gao, L. Wang, L. Zhou, W. Yang, and D. Shen. *Interactive medical image segmentation via point-based interaction and sequential patch learning*. arXiv preprint arXiv:1804.10481 (2018). [52](#)
- [112] H. B. Mann and D. R. Whitney. *On a test of whether one of two random variables is stochastically larger than the other*. The annals of mathematical statistics pp. 50–60 (1947). [55](#)



- 
- [113] J. West, D. Ventura, and S. Warnick. *Spring research presentation: A theoretical foundation for inductive transfer*. Brigham Young University, College of Physical and Mathematical Sciences **1** (2007). [58](#)
- [114] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. *Attention is all you need*. In *Advances in Neural Information Processing Systems*, pp. 5998–6008 (2017). [58](#)