

# Computational Methods for MRI-Guided and Powered Ferric Applicators: Modeling and Image Processing

by  
Wenhui Chu

A dissertation presented to the Faculty of the Department of Computer Science,  
College of Natural Sciences and Mathematics  
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy  
in Computer Science

Chair of Committee: Nikolaos V. Tsekos

Committee Member: Larry Shi

Committee Member: Christoph F. Eick

Committee Member: Aaron T. Becker

University of Houston  
December 2021

Copyright 2021, Wenhui Chu

## ACKNOWLEDGMENTS

Firstly, I would like to express my gratitude to Dr. Nikolaos V. Tsekos for his excellent work as my advisor. His patience, guidance, knowledge, and motivation were key factors to guarantee the successful outcome of my academic and research activities during my stay at the University of Houston as a Ph.D. student. Over the years, he has dedicated his support to helping me to cross the finish line.

Besides my advisor, I would like to thank the rest of my dissertation committee: Dr. Weidong Shi, Dr. Christoph F. Eick, and Dr. Aaron T. Becker for serving on my dissertation committee. I appreciate all of their valuable feedback.

Last but not the least, I would like to thank my family: my wife Fan and my daughter Priscilla, because without their support it would have been impossible to achieve this goal. To my parents, who provided me with all the tools and resources needed to accomplish this new chapter of my life. Finally, to the rest of my family for their help and support.

## ABSTRACT

Cardiac diseases are major causes of global mortality which are a consistent threat to the lives of people. With the development of left ventricle segmentation, the real-time MRI-based control of a ferromagnetic application for endovascular navigation with data sensing and feedback in cardiac was applied in recent years. In this work, we first propose three novel deep learning architectures called BNU-net, LNU-net, and IBU-net for left ventricle segmentation from short-axis cine MRI images. BNU-net is the batch normalized (BN) U-net, LNU-net is the layer normalized (LN) U-net, and IBU-net is the instance-batch normalized (IB) U-net. The architectures of BNU-net, LNU-net, and IBU-net have an encoding path for feature extraction and a decoding path that enables precise localization. BNU-net, LNU-net, and IBU-net have left ventricle segmentation methods: BNU-net employs batch normalization to the results of each convolutional layer, LNU-net applies layer normalization in each convolutional block, while IBU-net incorporates instance and batch normalization together in the first convolutional block. Our method incorporates affine transformations and elastic deformations for image data processing. Our dataset that contains 805 MRI images regarding the left ventricle from 45 patients is used for evaluation. The experimental results reveal that our approach accomplishes comparable or better performance than other state-of-the-art approaches in terms of the dice coefficient and the average perpendicular distance.

We then simulate a computational platform for preoperative planning and modeling of MRI-powered applicators inside blood vessels. This platform was implemented as a two-way data and command pipeline that links the MRI scanner, the computational core, and the operator. The platform first processes multi-slice MR data to extract the vascular bed and then fits a virtual corridor inside the vessel. This corridor serves as a virtual fixture (VF), a forbidden region for the applicators to avoid vessel perforation or collision. The geometric features of the vessel centerline, the VF, and MRI safety compliance ( $\text{dB}/\text{dt}$ , max available gradient) are then used to generate magnetic field gradient waveforms. Different blood flow profiles can be user-selected, and those parameters are used for modeling the applicator’s maneuvering.

# TABLE OF CONTENTS

|                                                                                                             |             |
|-------------------------------------------------------------------------------------------------------------|-------------|
| <b>ACKNOWLEDGMENTS</b>                                                                                      | <b>iii</b>  |
| <b>ABSTRACT</b>                                                                                             | <b>iv</b>   |
| <b>LIST OF FIGURES</b>                                                                                      | <b>viii</b> |
| <b>LIST OF TABLES</b>                                                                                       | <b>ix</b>   |
| <b>1 Introduction</b>                                                                                       | <b>1</b>    |
| 1.1 Previous and Current Work . . . . .                                                                     | 1           |
| 1.1.1 Cardiac Segmentation . . . . .                                                                        | 1           |
| 1.1.2 Image Segmentation . . . . .                                                                          | 4           |
| 1.1.3 CNN-based Segmentation . . . . .                                                                      | 8           |
| 1.1.4 RNN - Recurrent Neural Network . . . . .                                                              | 13          |
| 1.1.5 Segmentation Based on Combination of CNN and RNN . . . . .                                            | 13          |
| 1.1.6 GAN - Generative Adversarial Networks . . . . .                                                       | 14          |
| 1.1.7 Robot-assisted Technologies . . . . .                                                                 | 15          |
| 1.2 Motivation . . . . .                                                                                    | 16          |
| 1.2.1 Challenges . . . . .                                                                                  | 17          |
| 1.3 Contributions . . . . .                                                                                 | 18          |
| 1.4 Organization . . . . .                                                                                  | 19          |
| <b>2 Background</b>                                                                                         | <b>20</b>   |
| 2.1 Theoretical Background . . . . .                                                                        | 20          |
| 2.1.1 Cardiac Segmentation . . . . .                                                                        | 20          |
| 2.1.2 Robot-assisted Technologies . . . . .                                                                 | 26          |
| <b>3 Automated Segmentation of Left Ventricle</b>                                                           | <b>27</b>   |
| 3.1 BNU-net: a Novel Deep Learning Approach for LV MRI Analysis in Short-Axis MRI                           | 27          |
| 3.1.1 Fully Convolutional Neural Networks . . . . .                                                         | 27          |
| 3.1.2 The Structure of Network . . . . .                                                                    | 30          |
| 3.1.3 Using Batch Normalization in BNU-net . . . . .                                                        | 33          |
| 3.1.4 Applying Activation Function . . . . .                                                                | 34          |
| 3.1.5 Dataset . . . . .                                                                                     | 37          |
| 3.1.6 Experimental Studies . . . . .                                                                        | 38          |
| 3.1.7 Evaluation Metrics . . . . .                                                                          | 40          |
| 3.1.8 Segmentation Results and Comparison . . . . .                                                         | 41          |
| 3.1.9 Discussion . . . . .                                                                                  | 43          |
| 3.2 Two Deep Learning Approaches for Automated Segmentation of Left Ventricle in Cine Cardiac MRI . . . . . | 44          |
| 3.2.1 Methodology . . . . .                                                                                 | 44          |
| 3.2.2 Dataset . . . . .                                                                                     | 50          |
| 3.2.3 Model . . . . .                                                                                       | 50          |
| 3.2.4 Experimental Studies . . . . .                                                                        | 51          |

|          |                                                                                                                      |           |
|----------|----------------------------------------------------------------------------------------------------------------------|-----------|
| 3.2.5    | Segmentation Evaluation . . . . .                                                                                    | 52        |
| 3.2.6    | Discussion . . . . .                                                                                                 | 56        |
| <b>4</b> | <b>Simulations of MRI Guided and Powered Ferric Applicators for Tetherless Delivery of Therapeutic Interventions</b> | <b>58</b> |
| 4.1      | Methods . . . . .                                                                                                    | 58        |
| 4.1.1    | MRI-driven Devices . . . . .                                                                                         | 58        |
| 4.1.2    | Overview the VF Approach . . . . .                                                                                   | 59        |
| 4.2      | Theoretical Model . . . . .                                                                                          | 60        |
| 4.2.1    | Interpolation . . . . .                                                                                              | 60        |
| 4.2.2    | Calculating the Tortuosity . . . . .                                                                                 | 60        |
| 4.2.3    | Blood Velocity . . . . .                                                                                             | 61        |
| 4.2.4    | Different Type of Blood Flow . . . . .                                                                               | 61        |
| 4.2.5    | Velocity of Sphere . . . . .                                                                                         | 61        |
| 4.3      | Overview of MRI Gradients . . . . .                                                                                  | 62        |
| 4.3.1    | Control Module . . . . .                                                                                             | 62        |
| 4.3.2    | Gradients Control . . . . .                                                                                          | 63        |
| 4.3.3    | Magnetic Force . . . . .                                                                                             | 66        |
| 4.4      | Simulation and Results . . . . .                                                                                     | 67        |
| 4.4.1    | Time-varying Magnetic Fields . . . . .                                                                               | 68        |
| 4.4.2    | Visualization Toolkit . . . . .                                                                                      | 71        |
| 4.4.3    | Simulation . . . . .                                                                                                 | 71        |
| <b>5</b> | <b>Conclusion</b>                                                                                                    | <b>74</b> |
|          | <b>BIBLIOGRAPHY</b>                                                                                                  | <b>76</b> |

# LIST OF FIGURES

|    |                                                                                                                                                                                                                                                                                    |    |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1  | Convolutional neural network for image processing, eg., handwriting recognition [104].                                                                                                                                                                                             | 9  |
| 2  | Architecture of LeNet-5 : a convolutional neural network, here for digits recognition [152]. . . . .                                                                                                                                                                               | 9  |
| 3  | Recurrent Neural Network for cardiac image segmentation [113]. . . . .                                                                                                                                                                                                             | 13 |
| 4  | (A) A generative adversarial network structure. (B) Adversarial training for image segmentation [113]. . . . .                                                                                                                                                                     | 15 |
| 5  | (A) Generic architecture of convolutional neural networks (CNN). (B) Patch-based segmentation method based on a CNN classifier [113]. . . . .                                                                                                                                      | 22 |
| 6  | U-net architecture. White boxes represent copied feature maps. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The arrows denote the different operations [21]. . . . .                                             | 25 |
| 7  | (A) Generic architecture of fully convolutional neural networks (FCN) for segmentation (B) A schematic drawing of U-net [113]. . . . .                                                                                                                                             | 28 |
| 8  | The 3D U-net architecture. Blue boxes represent feature maps. The number of channels is denoted above each feature map [67]. . . . .                                                                                                                                               | 29 |
| 9  | Architecture of the BNU-net convolutional network. (a) The contraction path is responsible for feature extraction. (b) Batch normalization is performed after each convolution in the convolutional layer. . . . .                                                                 | 31 |
| 10 | Dropout neural net model. Left: A standard neural net with 2 hidden layers. Right: Applying dropout to the network on the left that generates a thinned net [109]. . .                                                                                                             | 33 |
| 11 | Activation function [108]. . . . .                                                                                                                                                                                                                                                 | 35 |
| 12 | The contour comparison based on MRI input. (a) Sunnybrook Image and corresponding label. (b) Perform affine transformations. (c) Perform elastic transformation to augment the training set. . . . .                                                                               | 39 |
| 13 | Segmentation predictions on raw inputs from the Sunnybrook dataset. . . . .                                                                                                                                                                                                        | 43 |
| 14 | The LNU-net architecture of the proposed fully convolutional network. Layer normalization is performed after each convolution in the convolutional layer. . . . .                                                                                                                  | 47 |
| 15 | Architecture of the IBU-net convolutional network. Instance normalization is applied in the first convolutional layer. Batch normalization is performed after each convolution in the convolutional layer. . . . .                                                                 | 49 |
| 16 | Examples of segmentation results on raw inputs from three conditions in the Sunnybrook dataset. The first row contains heart failure with infarction, the second row represents hypertrophy, the third row shows healthy patients. . . . .                                         | 55 |
| 17 | Some segmentation outputs by our methods. The solid lines represent the segmentation examples of the Sunnybrook dataset. We compare four different network methods, which are U-net, BNU-net, LNU-net, and IBU-net respectively. . . . .                                           | 56 |
| 18 | Results of $T_p = 100$ ms (blue) and $T_p = 200$ ms (red) of the simulation in the constant blood flow. (a) is a plot of the curvature of the path. (b) shows the constant blood flow. (c), (d), and (e) present the gradients generated by the MRI scanner for each axis. . . . . | 64 |

|    |                                                                                                                                                                                                                                                                                                 |    |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 19 | Results of $T_p = 100$ ms (blue) and $T_p = 200$ ms (red) of the simulation in the normal blood flow. (a) is a plot of the curvature of the path. (b) shows the normal blood flow. (c), (d), and (e) present the gradients generated by the MRI scanner for each axis. . . . .                  | 65 |
| 20 | Results of $T_p = 100$ ms (blue) and $T_p = 200$ ms (red) of the simulation in fast heart rate of blood flow. (a) is a plot of the curvature of the path. (b) shows the fast heart rate blood flow. (c), (d), and (e) present the gradients generated by the MRI scanner for each axis. . . . . | 66 |
| 21 | Results of $T_p = 100$ ms (blue) and $T_p = 200$ ms (red) of the simulation in the constant blood flow. (a), (b), and (c) show $dB/dt$ for each axis. The blood flow is 1 ml/s. . . . .                                                                                                         | 69 |
| 22 | Results of $T_p = 100$ ms (blue) and $T_p = 200$ ms (red) of the simulation in the normal blood flow. (a), (b), and (c) show $dB/dt$ for each axis. . . . .                                                                                                                                     | 70 |
| 23 | Results of $T_p = 100$ ms (blue) and $T_p = 200$ ms (red) of the simulation in the fast heart rate of blood flow. (a), (b), and (c) show $dB/dt$ for each axis. . . . .                                                                                                                         | 71 |
| 24 | Structure of Qt user interface . . . . .                                                                                                                                                                                                                                                        | 73 |



## LIST OF TABLES

|   |                                                                                                                                                                   |    |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 1 | Comparison between our proposed BNU-net model with ELU and BNU-net with ReLU . . . . .                                                                            | 37 |
| 2 | Output of the model and efficiency metrics results . . . . .                                                                                                      | 41 |
| 3 | Segmentation results on the Sunnybrook dataset, compared to the performance from the state-of-the-art methods . . . . .                                           | 42 |
| 4 | Dice mean with different normalization . . . . .                                                                                                                  | 52 |
| 5 | Output of the models and efficiency metrics results . . . . .                                                                                                     | 53 |
| 6 | Dice score and average perpendicular distance (APD) of segmenting the Sunnybrook dataset, compared to the performance from the state of the art methods . . . . . | 54 |

# 1 Introduction

Left ventricle (LV) segmentation is critical for clinical quantification and diagnosis of cardiac images. Motivated by the development of automated cardiac segmentation methods, we decided to derive the machine learning background for using normalization techniques in deep learning networks. Cardiac segmentation is important to the facilitation of diagnostic processes. With this in mind, we used different networks and compared them with ours to see which performs best. Ultimately, we found that our network solutions were better than those of other networks. The advent of real-time image guidance provides new opportunities in the field of interventional medicine [9]. We want to combine such magnetic actuation with a proper imaging modality, allowing for the tracking of MRbot in the body and opening the possibility for closed-loop servo control along pre-planned trajectories. So in our work, we used a method which is called magnetic resonance targeting (MRT), which has been demonstrated in vivo.

In this chapter, we present the previous and current work about segmentation, generative adversarial networks, and maneuvering ferromagnetic applicators. We also present the motivation for our work, showing the benefits and challenges of ventricle segmentation from short-axis MRI and interventional procedures in the human body. Finally, we describe all the proposed techniques for cardiac segmentation based on deep learning, intravascular interventions based on robot-assisted technologies, and the structure of this dissertation.

## 1.1 Previous and Current Work

In this research, we will focus on machine learning techniques in recent advances, especially on some difficult medical image segmentation tasks, including CNN-based segmentation and cardiac segmentation.

### 1.1.1 Cardiac Segmentation

In recent years, deep learning has become the most widely used approach for cardiac image segmentation. Because of the complexity of medical images, the noise of the images, and the high

variability of images, there has been quite a lot of research focusing on the field of medical images with numerous solutions published in recent years. It is complicated to segment the medical images, even for specialists.

Poudel et al. [1] presented a recurrent fully convolutional neural network for MRI cardiac image segmentation. This network learns image representations from the full stack of 2D slices and can leverage inter-slice spatial dependences through internal memory units. The network combined anatomical detection and segmentation into a single architecture. This approach showed how to reduce computational time, simplify the segmentation pipeline, and potentially enable real-time applications. They trained the network using two datasets, including the MICCAI 2009 challenge dataset. The results showed the network produced state-of-the-art results and could significantly improve the delineation of contours near the apex of the heart.

Tran [2] proposed a deep fully convolutional neural network architecture. His model is the first application of a fully convolutional neural network architecture for pixel-wise labeling in cardiac magnetic resonance imaging. The model is trained in a single learning stage from whole-image inputs and grounds truth to make an influence at every pixel. Based on multiple evaluation measures, the experimental results showed that the model is robust to outperform previous fully automated methods.

Ngo et al. [3] proposed a new semi-automated method that was processed with the combination of level set methods and deep belief networks. They presented a top-down segmentation method with the knowledge automatically learned from a manually annotated database. The use of deep belief networks could learn robust models with few annotated images. This approach showed the advantages of the network, which used several a priori facts about the object to be segmented (e.g., smooth contour, strong edges, etc.).

Queirós et al. [4] proposed a novel automatic 3D+ time left ventricle segmentation framework. The framework consists of three conceptual blocks to delineate both epicardial and endo contours through the cardiac cycle : (a) an automatic stack initialization followed by a 3D segmentation at the end-diastolic phase; (b) an automatic 2D mid-ventricular initialization and segmentation;

and (c) a tracking procedure. The experimental results showed the robustness, efficiency, and competitiveness of the proposed method both in terms of accuracy and computational load.

Hu et al. [5] proposed a novel and robust model which could improve the accuracy of automatic left ventricle segmentation on short-axis cardiac. They presented three techniques when using this segmentation algorithm: (a) for left ventricular outflow tract, they used a ray scanning approach; (b) a region-restricted technique was employed for epicardial contour extraction; and (c) in order to improve the dynamic programming to derive the epicardial boundary, an edge map with non-maxima gradient suppression approach was used. They used three training datasets obtained from the Sunnybrook Health Sciences Centre. The segmentation model showed better performance and improved accuracy than other state-of-the-art models.

Huang et al. [6] proposed a comprehensive approach to automatically segment the left ventricle from short-axis cine cardiac MR images. The authors stated that accuracy and automation are the two important criteria in improving cardiac image segmentation. They introduced edge detection, thresholding, image filtering, and mathematical morphology to do image processing. The authors stated that cardiac MR images achieved high accuracy after using these image analysis techniques.

A fully automatic left ventricle segmentation from 4D cardiac MR was proposed by Margeta et al. [7] as a machine learning approach using two layers of spatio-temporal decision forests. They also presented a method for context-aware MR intensity standardization and image alignment, supporting the idea of using the 4D cardiac MR to segment the images.

Liu et al. [8] proposed three key techniques to segment the left ventricle in short-axis cardiac MRI. (a) they presented a topological stable-state thresholding method to refine the endocardial contour; (b) an edge map with a non-maxima gradient suppression approach; and (c) in order to derive the epicardial boundary, a region-restricted technique was proposed and improved dynamic programming. They stated that this novel segmentation approach got better performance and had great potential in improving the accuracy of computer-aided diagnosis systems in cardiovascular diseases.

Zhao et al. [63] proposed a novel mathematical morphological edge detection algorithm. They

stated that it is not good to detect medical image edge-based based on gradient algorithm and template algorithm, while medical images have noises. This research showed that the algorithm is more efficient for medical image denoising and edge detection than other algorithms.

Çiğla et al. [64] proposed a graph theoretic color image segmentation algorithm. They stated that the image is represented by a weighted undirected graph, whose nodes correspond to over-segmented regions, which decreases the complexity of the algorithm. The irregular distribution of nodes could cause a bias towards combining regions with a high number of links. For each node, this bias is removed by limiting the links. These novel ideas performed faster than the traditional cut methods and got better segmentation results.

Xu et al. [65] proposed a threshold-based segmentation method. In their approach, the original image was pre-processed to enhance the image edge and filter noise. They used the threshold segmentation method to control the diffusion coefficient of the curve in the threshold set and used the improved fast marching method (FMM) to segment the image. This research showed that the threshold-based approach can outperform the traditional segmentation algorithms under certain conditions.

### **1.1.2 Image Segmentation**

Image segmentation is one of the most fundamental problems in image understanding and has been used in many areas, such as computer vision, medical imaging, and object detection. In general, image segmentation is the process of partitioning a digital image into multiple meaningful regions. Voxels (or pixels) in each segment should have some common signal characteristics. For example, the intensity of color consistency to show that same object of interest or same category that they belong to. There are different interpretations of image segmentation when it comes to specific application areas. For example, machine vision is often considered to be composed of two steps which are low-level vision and high-level vision [81]. Change detection based on remote sensing data is a prerequisite step that is used in detecting changes on the earth's surface and global environmental change [82]. In medical areas, automatic segmentation of medical images has played

a vital role in detecting diseases in patients' early stages. For example, in cardiac MRI analysis, the goal of image segmentation is to visualize, measure, and analyze the specific anatomical structure of the cardiac. The boundary between the organs and regions of interest (ROI) is often unclear [83]. It is easy for doctors to analyze the images accurately after the ROIs are segmented out.

Lankton et al. [84] proposed a model that allows any region-based segmentation energy to be re-formulated locally. The framework used local image statistics and evolve a contour based on local information. They applied a local region-based energy function rather than a global region-based energy function. Because using a standard global method was difficult to capture the objectives features, the framework localized the contours. It obtain accurate segmentation results based on choosing a good localization scale for images.

Tan et al. [85] proposed an approach called histogram thresholding fuzzy c-means hybrid (HTFCM) that could obtain a different application in computer vision and pattern recognition. This method was applied the histogram thresholding technique to get all possible uniform regions in the color images. Then the Fuzzy C-means (FCM) algorithm is applied to improve the compactness of the clusters forming these uniform regions. The experimental studies showed that the HTFCM approach could achieve better segmentation results and better cluster quality compared with other segmentation approaches.

Dollar et al. [86] presented a supervised learning algorithm for edge and object boundary detection. Edge detection is one of the most challenging tasks in computer vision. It is difficult since the decision for an edge can not be decided by low-level cues. The novel algorithm used all of the information, which are low, middle, and high, to choose where to put edges. This research proposed a novel supervised learning algorithm for edge and object boundary detection, which is called Boosted Edge Learning (BEL). In an image, a decision of an edge point is made independently at each location. This model can extract the discriminative category features across different scales. The algorithm was tested on various datasets including the Berkeley dataset and the results obtained are very good.

Nguyen et al. [87] proposed an interactive framework based on the developed continuous-domain convex active contour model. This model showed robustness to user inputs and different initializations. The algorithm produced an accurate and smooth boundary contour with simple user inputs and handle topology changes. The experimental results outperformed the state-of-the-art image segmentation algorithms.

Ju et al. [88] proposed an image segmentation algorithm based on adaptive edge detection and an improved mean shift. Mean-shift is a nonparametric clustering method based on kernel density estimation for feature space analysis. An improved fast mean shift algorithm was proposed in this research. According to the method, detecting an edge was a fundamental step for most computer vision applications such as MRI feature extraction and an adaptive threshold algorithm was applied in edge detection. This research used the concept of scaling factor to represent the expansion degree of the enlarged mean-shift vector. The experimental results got better performance compared with several state-of-the-art algorithms.

Cour et al. [89] proposed a multiscale image segmentation algorithm in parallel, without iteration, to capture both coarse and fine level details. This algorithm is computationally efficient, allowing to segment of large images. The segmentation algorithm works across the gross scales simultaneously and can capture image structure at increasing large neighborhoods.

Wang et al. [90] proposed an image segmentation method that considered the local image information by describing it as a novel local signed difference (LSD) energy, which possessed both local separability and global consistency. The LSD energy term is integrated into an objective energy functional, which is minimized via a level set evolution process. The experiment results showed that the proposed method improved the accuracy and efficiency compared with the state-of-the-art approaches.

Comaniciu et al. [91] proposed a mean shift method, which is a nonparametric technique for the analysis of complex multimodal feature space and to delineate arbitrarily shaped clusters in it. The core computational module of this method was an old pattern recognition procedure, which is called the mean shift. Experimental results illustrated the excellent performance in image segmentation.

Melkemi et al. [92] presented a distributed image segmentation algorithm structure composed of a set of segmentation agents and a coordinator agent. To get a sub-optimal segmented image, every segmentation agent performed the iterated conditional modes methods. This method increased the accuracy and efficiency of the segmentation results but has high time complexity.

Arbelaez et al. [93] proposed a novel design for region-based object detectors that integrated efficiently top-down information from scanning-windows part modules and global appearance cues. His research focused on challenging articulated categories such as humans and other animals, and their detectors produce a class-specific score for bottom-up regions. He evaluated the approach on the PASCAL segmentation challenge and achieved good performance.

Chen et al. [94] proposed a model which was called artificial co-evolving tribes and applied it to solve the image segmentation problem. During the process, the individuals who made up the tribes affect communication cooperatively from one agent to the other to increase the homogeneity of the ensemble of the image regions.

Arbelaez et al. [95] proposed state-of-the-art algorithms to solve contour detection and image segmentation problems. Their segmentation algorithm consists of transforming the output of any contour detector into a hierarchical region tree. It can help to reduce the problem of image segmentation to that of contour detection. The experimental studies showed that their contour detection and segmentation methods outperform other research groups' algorithms. The automatic segmentation methods can be refined by user-specific annotations.

Huang et al. [96] proposed a novel segmentation algorithm based on artificial ant colonies (AC). It showed that the self-organization of ants was similar to neurons in the human cardiac in many respects. In this model, each ant could memorize a reference object, which would be refreshed when it found a new target. The behavior of an ant was affected by the neighbors and the cooperation between ants was performed by exchanging information through pheromone updating. It is similar to neurons in the human cardiac. They presented an extended model for image segmentation. In their model, each ant could memorize a reference object, which would be refreshed when it found a new target. To evaluate the similarity between the reference object and the target, a



fuzzy connectedness measure was used. The experimental results showed that the new algorithm is insensitive to noise and can preserve the detail of the object.

Han et al. [97] proposed a fuzzy ant colony algorithm (ACA). The algorithm was inspired by the food-searching behavior of ants. In this research, image segmentation was used by fuzzy clustering. There are three features, which are gray value, gradient, and neighborhood of the pixels, that are extracted for the searching and clustering process. The algorithm has improved on enhancing the heuristic function to accelerate the searching process.

In the past few years, deep learning models that utilized different types of deep artificial neural networks have been successfully used in speech recognition, natural language processing, visual recognition, image segmentation, image recognition, and image classification. They have achieved great success in the field of semantic segmentation. The models have been produced good results for solving many difficult segmentation tasks in medical areas. In this research, we will focus on developing a new deep learning model to solve left ventricle cardiac segmentation problems.

### **1.1.3 CNN-based Segmentation**

A convolutional neural network (CNN) is a deep learning architecture, in which the connectivity pattern between its neurons is inspired by the organization of the animal's visual cortex. CNN has excellent feature extraction and was a particular kind of feedforward neural network. In 1989, LeCun [103] originally proposed a CNN model which was called LeNet-5. See Figure 1.

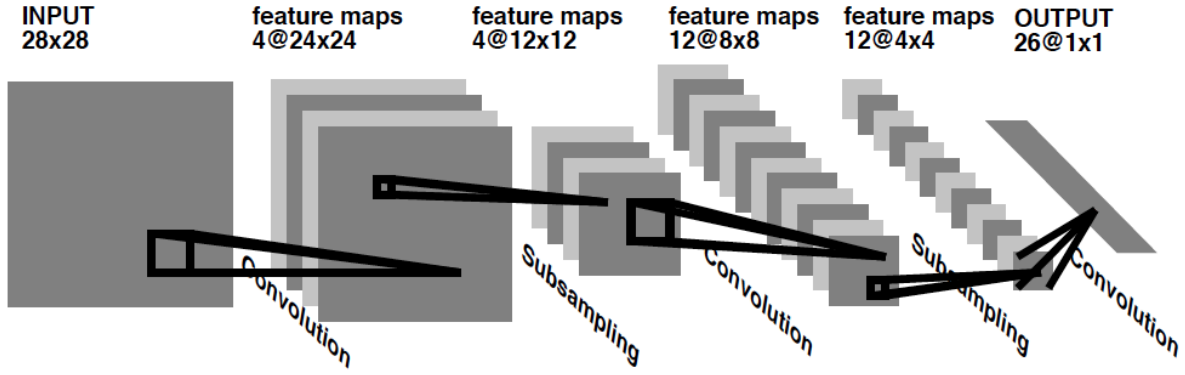


Figure 1: Convolutional neural network for image processing, eg., handwriting recognition [104].

This model was used for the recognition of handwritten digits and improved in [152]. See Figure 2.

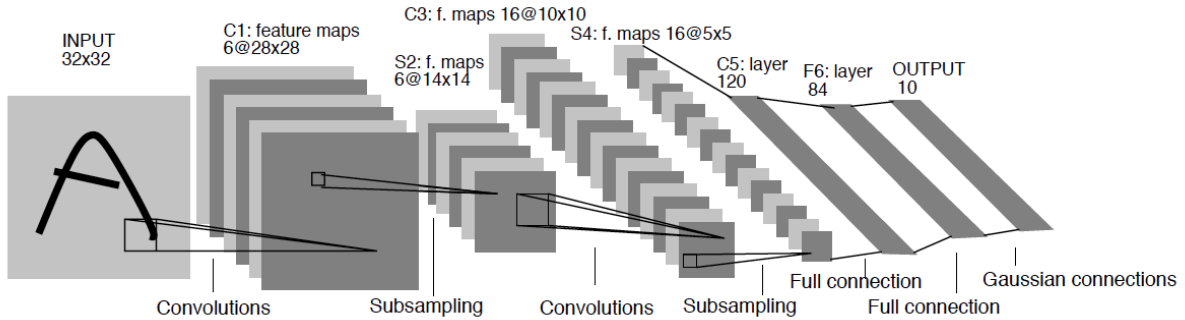


Figure 2: Architecture of LeNet-5 : a convolutional neural network, here for digits recognition [152].

Compared with other networks, LeNet-5 network has a multi-layer artificial neural network and can be trained with the backpropagation algorithm [105]. LeNet-5 is the first successful application of CNNs and was the first work to show the practical need for an improvement of conventional neural networks. LeNet-5 applies convolution operations and replaces matrix multiplications in certain layers. The network uses convolution operations to extract local features and combines them to get precise localization. When creating and training a neural network, there are several steps that we can follow. (a) choose data set (training/validation/testing); (b) specify the objective function to

use; (c) determine an optimization procedure to train the model. For example, the images contain objects of interest to be segmented in the input and the output of the model is the predicted segmentation probability map. In order to train the model, stochastic gradient descent (SGD) is used as the optimizer to update the weights of the model. It is common to use cross-entropy as an objective function that is defined on the ground truth and output results.

The CNN architecture includes several building blocks, such as the input layer, output layer, and hidden layers. Each hidden layer is made up of a specific operation, such as pooling, convolution, and activation. The input image is sent to the input layer and the number of neurons in this layer is the pixel of the input image. Most recently, CNN has been applied to advance the state-of-the-art successfully on various domains, including segmentation tasks, object detection, image classification, and image computing.

A convolutional layer is a fundamental component of the CNN architecture, which is composed of a stack of mathematical operations. It receives the input data and performs feature extraction. The pooling layer behind the convolutional layer selects the feature maps, simplifying the computational complexity of the entire network. It provides a down-sampling operation that reduces the in-plane dimensionality of the maps, precise segmentation for margin detection, and this is especially important for clinical applications. Its aim is to estimate the object of interest in an image. With the large increase in the proposed network structure, deep learning-based approaches have been improved to obtain good dice scores and accurate results in medical image segmentation. The major medical image understanding tasks are object localization, which used CNNs for cardiac segmentation. CNN shows good performance in image detection and segmentation with feature extraction and feature capabilities.

Many research groups [24][77][78][100][101] have proposed new methods to overcome the difficulties encountered in the training CNNs since 2006.

Krizhevsky et al. [24] trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images and won the ILSVRC-2012 computer vision competition. In order to make training fast, the authors used non-saturating neurons and proposed a high-optimized

GPU implementation of 2D convolution. They used the dropout regularization method to reduce overfitting in the fully-connected layers. Their results showed that a large, deep convolutional neural network is capable of achieving good results.

Ciresan et al. [78] proposed a deep and simple neural network that is used on handwritten digit recognition. They used back-propagation for plain multi-layer perceptrons which got a very low error rate on the MNIST handwritten digits benchmark.

Ren et al. [100] proposed real-time object detection with region proposal networks (RPN). In this work, they introduced a region proposal network that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. An RPN is a fully convolutional network that is to generate high-quality region proposals. The experimental studies revealed that their architecture achieved state-of-the-art object detection accuracy.

Andrej et al. [101] proposed a model that generated natural language descriptions of images and their regions. They combined bidirectional recurrent neural networks over sentences, CNN over image regions, and a structured objective that aligns the two modalities through a multimodal embedding. The multimodal recurrent neural network architecture applied the inferred alignments to learn to generate novel descriptions of image regions. The results showed that the model achieved state-of-the-art accuracy on three datasets.

Ciresan et al. [106] proposed a special type of deep artificial neural network as a pixel classifier. Each pixel is classified by using the trained neural network on a patch surrounding the pixel. The image in the window is segmented in a sliding window fashion.

Brebisson et al. [107] proposed automatic segmentation magnetic resonance (MR) images of the human cardiac into anatomical regions. This method presented a 2D/3D input image and applied a multi-scale schema to enforce the spatial consistency of the 3D whole cardiac MRI segmentation.

Training CNN-based segmentation is one of the most difficult problems in deep learning. Many solutions and strategies are proposed to ease the training of CNN. Several optimization methods are used for CNN, such as optimization and transfer learning.

**Optimization** A feedforward neural network is an artificial neural network, which is to approximate some function so that with a certain kind of input the network could generate the expected output. Because the function evaluates the information flow through the intermediate computations and finally to the output, we called it feedforward. There are no feedback connections in which outputs of the model are fed back into itself. Thus, gradient descent is used to optimize nearly all deep learning problems through propagating the gradient to update the parameters of each layer from the output layer to the input layer. They are called back-propagation, an iterative optimization procedure.

Traditionally, in machine learning networks, large training data sets improve the recognition ability, but also increase the computational complexity. Different from optimization for traditional algorithms that use all the training examples simultaneously in a large batch, deep learning models use a mini-batch optimization strategy. On each iteration, the mini-batch optimization strategy updates to the parameters based on an expected value of the cost function. They are called mini-batch stochastic gradient descent (SGD). There are two reasons to use a smaller batch size: (a) make it easier to fit one batch worth of training data in memory and (b) offer a regularizing effect and lower generalization error. Gradient descent has been regarded as slow. In a deep learning network, it is very important to improve the speed of the training process without hurting convergence for SGD. Generally, it is no guarantee to have a fast and effective convergence when using plain vanilla SGD. Firstly, the learning rate was set the same for all dimensions without considering each distribution to the overall cost in different ways. Secondly, it is difficult to choose an appropriate learning rate. A small learning rate leads to slow convergence, while a large learning rate will cause the loss function to fluctuate around the minimum. Lastly, how to avoid getting trapped in a local minimum where the gradient is close to zero in all directions. Several solutions have been developed. For example, using a second-order derivative of the cost function to guide and speed up the gradient descent. We can set different learning rates for each dimension of parameters. Several SGD variants [102] were developed in the last couple of years, such as AdaGrad, SGD with momentum, RMSProp, Adadelta, and Adam.

#### 1.1.4 RNN - Recurrent Neural Network

Recurrent neural network (RNNs) [1] are artificial neural network which are specialized for processing sequential data  $x_0, x_1, x_2, \dots, x_n$ , such as cine MRI and ultrasound image sequences. See Figure 3.

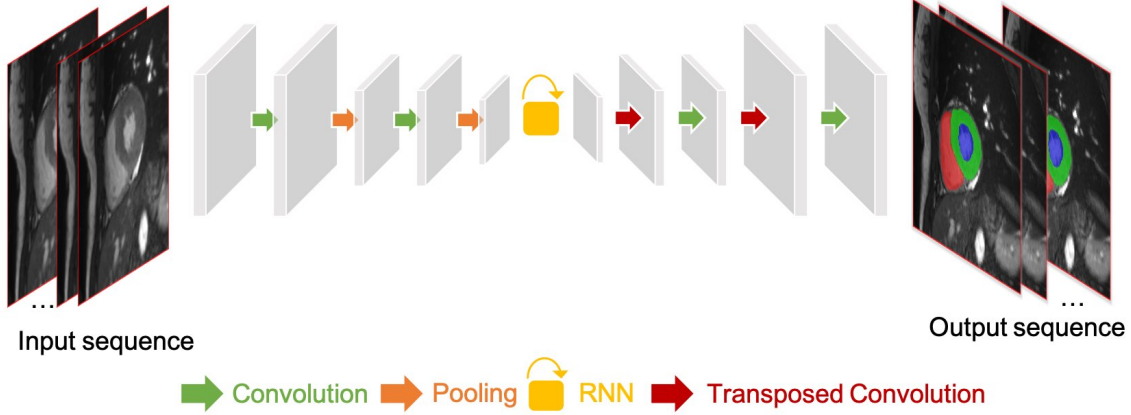


Figure 3: Recurrent Neural Network for cardiac image segmentation [113].

It takes an input sequence, connects hidden units, and produces an output sequence. The yellow block with a curved arrow means an RNN module. An RNN can ‘memorize’ and use the past knowledge information to make a present decision. An RNN takes the first input image, learns the information to make a prediction, and ‘memorize’ the information which is used to predict the next image. Several research groups are using RNN to solve natural language processing problems [110]-[113]. The two widely used methods in RNN are LSTM [115] and gated recurrent unit [116].

#### 1.1.5 Segmentation Based on Combination of CNN and RNN

In the previous two sections, we introduced that both CNN and RNN can be used in image segmentation. There are several steps for a 3D CNN-based segmentation model: (a) treat 3D image volume as a stack of 2D slices; (b) apply 2D segmentation models on the 2D slices, and (c) combine the 2D segmentation results. There are two drawbacks when using CNN to do 3D image segmentation: (a) lose the spatial information along the third direction and (b) more samples and parameters are

needed in 3D convolutional kernels. Because RNN can reserve the temporal-spatial information, we combine CNN and RNN on 3D medical image segmentation. Several research groups used this method to combine RNN and CNN together [138][139][140].

#### **1.1.6 GAN - Generative Adversarial Networks**

Goodfellow et al. [98] proposed a generative adversarial network (GAN), which is for image synthesis from noise. A GAN is a deep generative model which learns to model the data distribution of real data. See Figure 4(A). GAN is a class of machine learning framework that consists of a generator and a discriminator. The generative model is pitted against a discriminator. The two neural works contest with each other during the training process. The generator takes random noise and produces fake images, while the discriminator aims to figure out the real images from the fake images. This procedure is called adversarial training. This network can be used to image segmentation areas. From Figure 4(B) we can see that the generator is replaced by a segmentation model, the discriminator tries to distinguish the ground truth from the generated images.

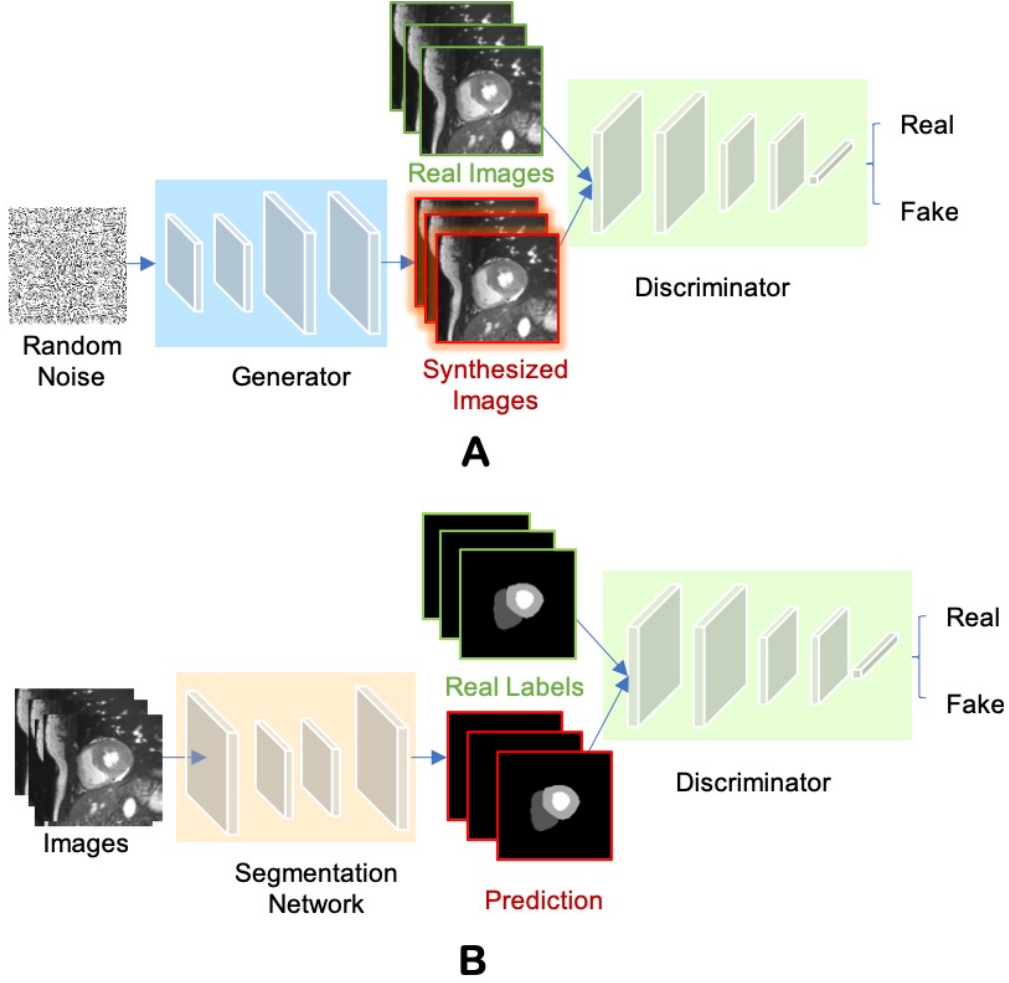


Figure 4: (A) A generative adversarial network structure. (B) Adversarial training for image segmentation [113].

### 1.1.7 Robot-assisted Technologies

Minimally invasive surgery (MIS) encompasses a specialized type of surgical method that allows surgeons to dramatically reduce patient discomfort and recovery time. Many researchers make efforts in micro-and nanorobotics, which have evolved to encompass platforms and technologies aimed at improving interventional procedures in the human body through the use of navigable or actuated magnetic spheres.

Magnetism is a prioritized choice for moving such micro-scale robots. The ferromagnetic core



embedded in the micro-device will provide the driving force from an MRI system. It is difficult to offer minimally invasive surgery, because of continuous cardiac motion. It is based on magnetic actuation and is known as Magnetic Resonance Navigation (MRN). MRN has drug-loaded micro-carriers embedded in a magnetic sphere, which moves through the vessel network from the injection point to the targeted region. The targeting of therapeutics to tumors has a great advantage in increasing the efficiency of treatments while reducing their secondary toxicity effects [13].

## 1.2 Motivation

In the United States, cardiovascular disease is the primary cause of death for both males and females [14]. One of the parameters that cardiologists examine in the diagnosis of heart disease is the amount of blood ejected by the left ventricle [15]. Physicians use Magnetic Resonance Imaging (MRI) scans to obtain relevant images of the cardiac areas to assess the structural and functional features for cardiovascular diagnosis and disease management in a non-invasive manner [15]. The main indicators of cardiovascular disease are the left ventricle (LV) end-systolic volume (ESV), the end-diastolic volume (EDV), and the ejection fraction (EF) [16]. The segmented contour of the left ventricle has been crucial in determining ESV and EDV. Therefore, coherent and precise segmentation of the LV from MRI images is critical to the precision of identification of ESV, EDV, and EF, which is crucial to determine cardiac disease in a non-invasive manner.

Using the magnetic field gradients of the MR scanner to propel a ferromagnetic sphere in small vessels such as capillaries for delivery of therapeutics is rapidly evolving due to cost-effectiveness and patient management. Therefore, our goal is to simulate MRI-guided and powered ferric applicators for tetherless delivery of therapeutic interventions. Our studies have shown that it is feasible to design an MRbot with propulsion forces considering the characteristics of the human cardiovascular system. This tracking technique is integrated into propulsion and a PID controller successfully, which allows real-time automatic navigational control of an MRbot in the blood vessels.

### 1.2.1 Challenges

Some challenges in the automatic left ventricle segmentation must be solved before it can be accepted in a clinical setting. Firstly, for cardiovascular disease, it takes physicians several minutes to diagnose a patient’s condition and the obtained results are not easily reproducible [17]. So, this work is a tedious and long task, with intra- and inter-observer variability, and poor reproducibility. Secondly, it is not easy to segment the left ventricle because of weak edge information. Thirdly, the sharpest boundary between the blood pool and myocardium is not necessarily the endocardium. Fourthly, during the cardiac cycle between systolic and diastolic phases, the shape of the left ventricle varies significantly. Lastly, there is the overlap of the surrounding background regions and the intensity distributions of cardiac structures. In recent years, the development of automated cardiac segmentation methods based on magnetic resonance imaging datasets is a crucial step to facilitate this cumbersome diagnosis process [18]. Using an improved automatic process to determine heart parameters and function can lead to a quicker, coherent diagnosis and generate a repeatable diagnostic process. While research over the past decade has addressed some of the above-mentioned technical difficulties in achieving tangible progress on automatically generated ventricle segmentation from short-axis MRI, the corresponding automated segmentation contours still have to be significantly improved for clinical use [17]. In addition, assessment of prior studies on small benchmark datasets, which may not reflect real-world variability in image resolution and heart physiological and functional features across locations, institutions, and populations, are restricted in scope [19]. We demonstrated an enhanced technique for automatic left ventricle segmentation in MRI pictures.

MRI images of cardiac diseases are an important reference for the diagnosis of heart diseases. Manual segmentation of MRI images requires plenty of time and the additional expenses of employing experts, which could be a valuable resource for more patients. In addition, this form of segmentation’s accuracy cannot be fully guaranteed. With the development of computational units, neural network scans can be applied in almost every aspect of our daily life. Medical care is also an important application, especially for image classification, image segmentation, and objective

detection. It is believed that image processing is more efficient and accurate with the help of neural networks.

For the ferric sphere, one of the key elements to consider when analyzing the use of intra-operative guidance is to fit a virtual corridor inside the vessel. We use the vessel centerline data extracted from the 3D vascular model produced by 3D angiography images. In the present work, we use MRI to generate a safe zone, which is called a safe corridor, in real-time inside the cardiac.

### 1.3 Contributions

For the left ventricle cardiac segmentation, we think a more accurate segmentation will be helpful for the diagnosis of heart disease, and this will save more lives in the future. Because accurate segmentation is the very first step in the evaluation of cardiac function, there have been some important studies on image segmentation for the left ventricle cardiac area in the past several decades [20]. In this research, we focus on pixel classification methods with U-net [21], which is a neural network widely used for medical applications. The U-net will be improved with different normalization methods, and we will compare their effects on results. In the left ventricle segmentation study, normalization is applied for deep learning for more accurate and efficient training. We would like to compare the effects of several approximation normalization methods with U-net in the application of left ventricle cardiac image segmentation. We first present the method to augment the original images and ground truth deformation. We then present three novel CNN architectures:

- The BNU-net (“Batch-Normalization-U-net”) that leverages the power of a fully convolutional neural network. It is designed to process 2D MRI images as inputs and produce a labeled slice as output. This work modified and improved the U-net architecture [21] to make it work with fewer training images and improve performance. The results of the performance comparison show that BNU-net gets a better dice score when compared with the original U-net.
- The LNU-net (“Layer-Normalization-U-net”), a network that applies layer normalization in

each convolutional block and is based on an exponential linear unit (ELU). It shows a minimal improvement of the proposed network in terms of the dice mean, average perpendicular distance, with similar performance in terms of sensitivity score.

- A new network called IBU-net which is based on instance normalization and batch normalization is effective for improving dice score in image segmentation tasks. To increase the dice mean of the medical segmentation problem, the ELU as an activation function was selected instead of ReLU in the original U-net.

For robot-assisted interventions technologies, we describe a computational approach that uses different blood flow to model the MRbot access corridor. The MRbot can be safely maneuvered for visual servoing on the Qt framework.

## 1.4 Organization

The remainder of this work is organized as follows:

Chapter 2 reviews the background and related work for the studied problem.

Chapter 3 describes the cardiac segmentation background and the proposed solutions based on short-axis MRI images.

Chapter 4 presents the minimally invasive surgery and solutions based on improving interventional procedures in the human body through the use of actuated magnetic spheres.

Finally, we present the conclusions of this work in Chapter 5.

## 2 Background

In this chapter, we introduce the general concepts, notations, terminologies, and the related work used in this dissertation.

In cardiac segmentation, we present the foundation of the U-net proposed by Ronneberger et al. [21] and the key ideas for cardiac segmentation based on short-axis MRI images.

In robot-assisted technologies, many researchers [10]-[12][141-151] chose micro-scale robots for minimally invasive surgery. Dahmen et al. [141] proposed path planning algorithms that worked on MRI data. The tracking algorithm would deliver position feedback for the ferromagnetic objects. By integrating path-planning methods and real-time tracking, the MRI system would provide this new functionality for controlled interventional targeted therapeutic applications. Li et al. [142] developed robotic software platforms that used MRI for feeding back information to a controller, which is used for navigating and controlling a wireless application in the pre-defined path of the blood vessels.

Belharet et al. [144] proposed a real-time MRI-based control of ferromagnetic application for endovascular navigation. They developed software that allowed to navigate the application along a 2D path in pulsative flow. The results showed the software was effective in the closed-loop control system.

### 2.1 Theoretical Background

#### 2.1.1 Cardiac Segmentation

Deep learning, as a branch of machine learning, consists of a hierarchical representation, which can be applied to a large array of problems. It uses a deep neural network to extract features from a large-scale dataset and simulates the learning process of the human cardiac. In deep learning networks, each neuron can be regarded as a small information-processing unit. The entire neural network is composed of many neurons. Today, cardiac segmentation of two-dimensional medical images is vital for medical image analysis. During the last decade, machine learning and deep

learning have been applied to understand images and segmentation, and have proven versatile and simplify the development of medical segmentation tasks compared to traditional methods. Among these works, a large number of these methods are used for left ventricle segmentation. The objective of ventricle segmentation is to delineate the endo- and epicardium of the LV and RV. Clinical indices need to get a segmentation map. Additionally, research groups stated that segmentation maps are essential for 3D+ time-motion analysis [135], 3D shape analysis [136], and survival prediction [137].

Its practical implementation in the clinical realm has been limited since well-labeled medical data is harder to obtain. However, the recent development of deep learning models has shown methods that could perform well with fewer quantities of data. It is an essential step to clinical cardiac contractile function quantification when doing left ventricle segmentation in short-axis sequences [17]. According to the statistics, manual segmentation by radiologists is time-consuming and inefficient work which takes about 20 minutes for the MR sequence of one subject [17]. The region based on segmentation methods include clustering [71], split and merge [72], thresholding [73], probabilistic atlas [74], and Gaussian-mixture model [75].

One of the breakthroughs that enabled the exploration of deep learning architectures is called convolutional NN (CNN).

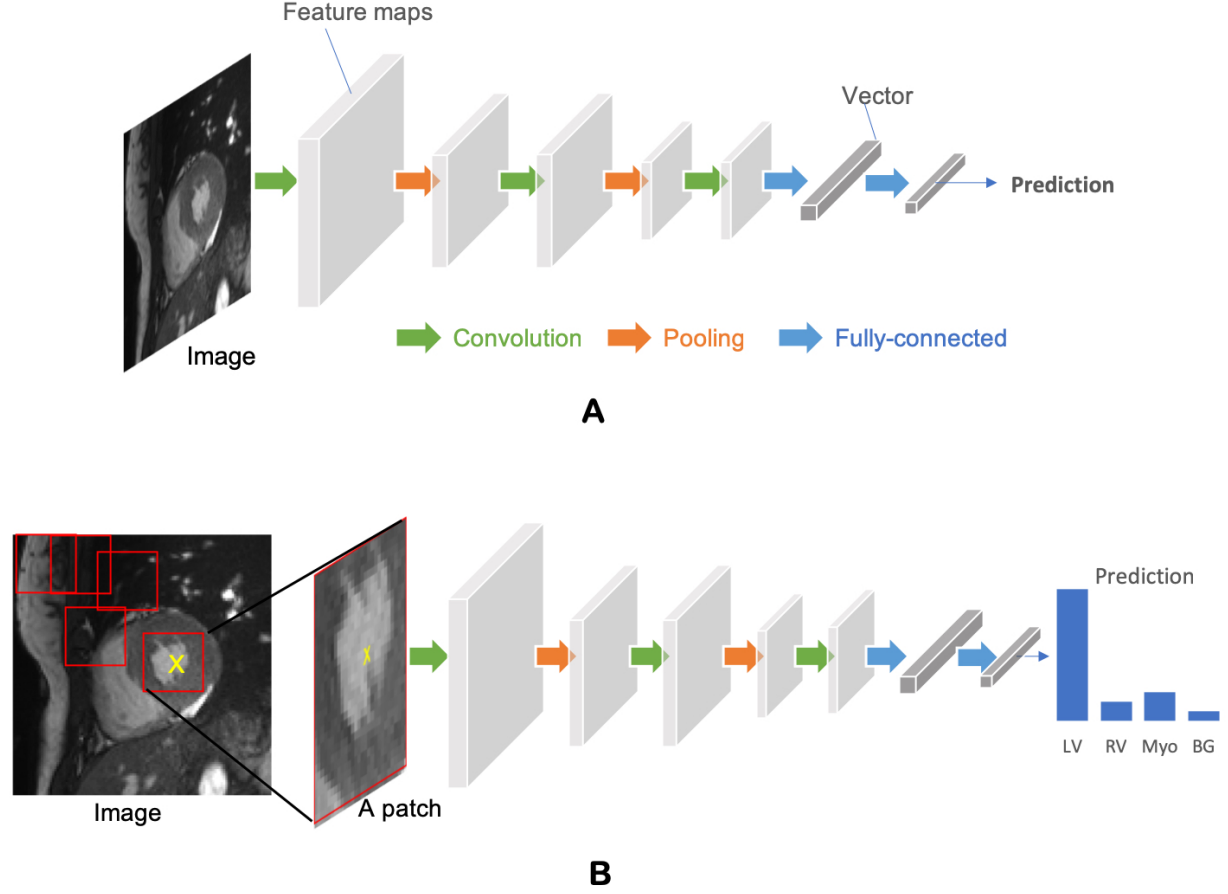


Figure 5: (A) Generic architecture of convolutional neural networks (CNN). (B) Patch-based segmentation method based on a CNN classifier [113].

CNNs are types of deep learning architectures, have been widely used for image classification tasks. A standard CNN consists of a convolutional layer, a pooling layer, and a rectified linear unit (ReLU). See Figure 5(A).

A CNN takes a cardiac MR image as input, learning features through convolutional layers, and pooling layers. The feature maps are flattened and reduced into a vector (image classification or object localization or patch-based segmentation) through fully connected layers. CNN has been applied to advance the state-of-the-art on objective detection, image classification, and segmentation tasks. From the proposal of the CNN to the current wide application, it is a classic model

produced by the combination of image processing and deep learning technology. Ciresan et al. [106] proposed that CNNs for image classification can also be employed for image segmentation without major adaptations to the network architecture. See Figure 5(B).

This method needs each image divided into patches, uses the network to train the image, and predicts the class label of the center pixel for every patch (yellow cross). The network is used for classification with patches in different patch locations and getting a pixel-wise segmentation map for the whole image. There is one disadvantage to using this approach, which is the network has to be deployed for each patch, and will be a lot of redundancy due to overlapping patches in the image.

To solve the computational cost problem for segmentation, Avendi et al. [114] used objective localization to estimate the bounding box of the object of interest in an image. The bounding box was then used to crop the image. CNNs are supervised training models which are trained to learn hierarchies of features automatically and perform classification robustly. Using deep learning models are proved to solve different image segmentation problems in recent years. New architectures of CNNs such as GoogLeNet [22], VGG-Net [23], Image-Net [24], and Res-Net [25], have made the CNN standard for classification in many applications [2].

Szegedy et al. [22] proposed a deep convolutional neural network called “GoogLeNet”. One of the major approaches to increase the performance of neural networks is to increase their size. While the bigger size of the neural network corresponds to a larger number of parameters, networks will be more prone to overfitting. If the size of the network increases, the computational resources will also increase. The authors designed the network which allowed for increasing the depth and width of the network while keeping the computational budget constant. This approach showed the improved utilization of the computing resources inside the network.

Simonyan et al. [23] proposed a very deep convolutional neural network (VGG) of increasing depth using an architecture with small (3x3) convolution filters. The results showed the network performed better than other state-of-the-art results.

He et al. [25] proposed a residual learning framework and trained the network easily. They



designed the framework that is substantially deeper than those used previously. They proposed a different approach from the traditional CNN network, reformulating the layers as learning residual functions about the layer inputs. This approach showed how residual net could be used to recognize images and presented a solution based on increasing depth that can be modified in the network.

Zhuang et al. [117] proposed a comparison of a group of methods [118][119][120][121][122][123] for the heart segmentation that has been experimented with and evaluated. Xu et al. [125] demonstrated 3D FCN to detect ROI for segmentation, focused on the anatomically relevant regions, and showed the effectiveness for heart segmentation.

Zreik et al. [123] gave a two-step LV segmentation process. A bounding box that was described in [124] was used for LV segmentation. The bounding box used a patch-based CNN followed by a voxel classification.

It is challenging to make a voxel-wise tissue classification if there is the image quality is low. In the last couple of years, research groups [126][127][128][129][130] used a deformable model to segment the LV in 2D images. [128][130] used pipeline which localized the ROI through the transformation of a bounding box and segmented the target structure. It has increased the robustness of the model and greatly reduced the search region when doing segmentation.

Although these previous studies have achieved good performance, they still lack accuracy, which can prevent them from effectively utilizing the left ventricle in clinical areas [18]. Ronneberger et al. [21] introduced a deep learning method called U-net, which is used for biomedical image segmentation, building upon the field of image classification and training strategy. See Figure 6.

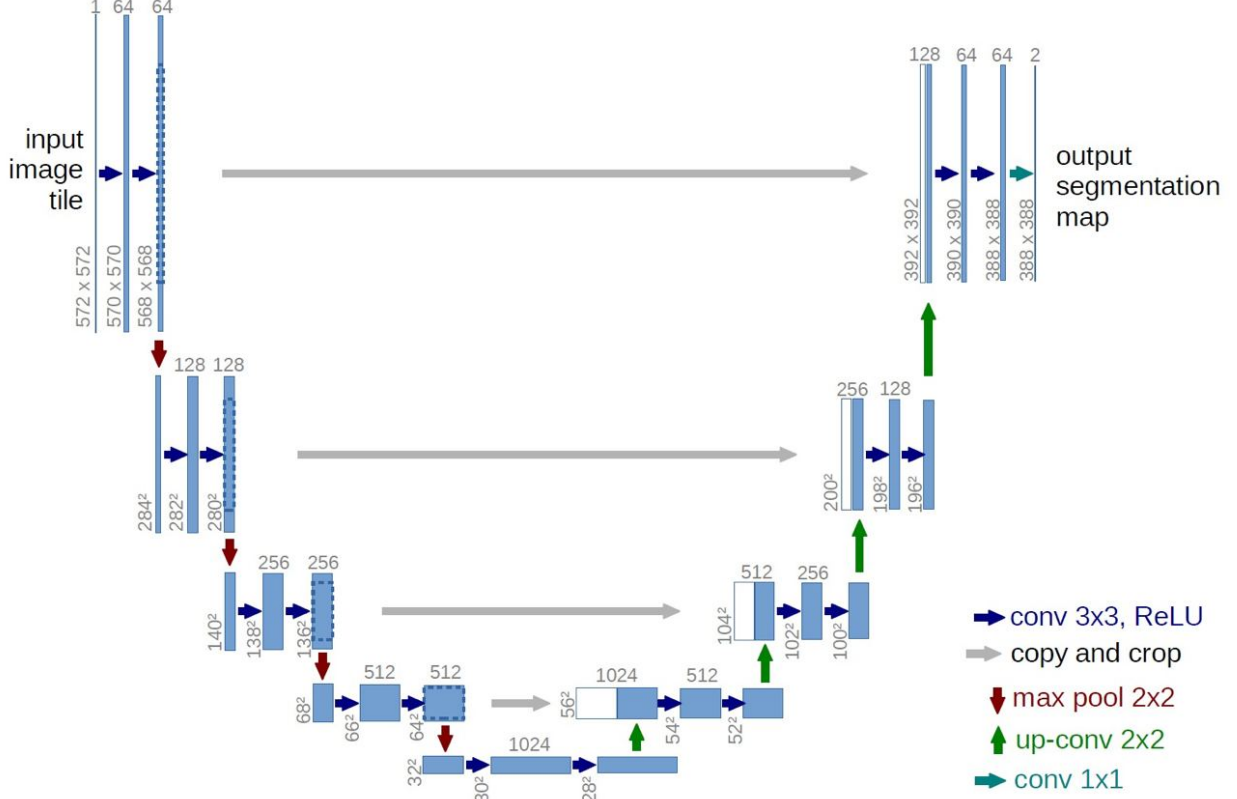


Figure 6: U-net architecture. White boxes represent copied feature maps. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The arrows denote the different operations [21].

The network relies on the use of data augmentation and expands the training dataset which makes samples more efficient. The architecture of this network is similar to a U-shape and consists of a contracting path to feature contraction and a symmetric expanding path that enables precise localization of biomedical objects [2].

Ronneberger’s work shows that little training data available can still get segmentation after applying elastic deformations to the available training images. Data augmentation plays an essential role in biomedical segmentation since elastic deformation used to be the most common variation in tissue and realistic deformations can be simulated efficiently [21].

After collecting and labeling large-scale datasets, CNN is one of the most effective models to

segment the images. Two research groups [131][132] have shown that it is capable to segment the images accurately without post-processing. Smistad et al. [131] proposed an efficient CNN method by performing real-time classification and segmentation. Leclerc et al. [132] performed to investigate the importance of the data size for segmentation of LV in 2D images using U-net. They demonstrated U-net benefits from a large number of datasets and got high accuracy.

### **2.1.2 Robot-assisted Technologies**

For robot-assisted technologies, the magnetic field gradients of the MRI scanner are used to power and maneuver ferromagnetic applicators for accessing sites in the patient’s body via the vascular network. Navkar et al. [9] describe a new method in the field of interventional medicine which provides real-time guidance for robot-assisted interventions. Magnetic resonance targeting (MRT) is to provide the driving force, which moves tiny tetherless therapeutic entities to the targeted region. It is based on using the magnetic field gradients of an MRI scanner, used for signal spatial encoding and image generation, to navigate micro-device through the cardiovascular system.

The above-mentioned clinical paradigms have an important role in robotic technology. The challenge is that with MRbots the gradients are also used for propulsion and there is a priority conflict: the selected vascular path can be safely maneuvered, which in turn requires continuous MRbot tracking and path imaging [56].

### 3 Automated Segmentation of Left Ventricle

#### 3.1 BNU-net: a Novel Deep Learning Approach for LV MRI Analysis in Short-Axis MRI

##### 3.1.1 Fully Convolutional Neural Networks

Although CNN as patch-wise classifiers for image segmentation task achieves good segmentation result, CNN has problems when spatial information would be abandoned during convolutional and pooling layers operation which will impact the segmentation accuracy. In order to solve this problem, the idea of a fully convolutional neural network (FCN) was first introduced by Long et al. [79] for image segmentation. FCNs are a special type of CNNs that do not have any fully connected layers. FCN is designed to have an encoder and decoder structure, which can take input of the arbitrary size and produce the same output image size.

Given an input image, the encoder first transforms the input into a high-level feature representation for feature extraction. The encoder is to reduce resolution and increase the depth of the network. The decoder interprets the feature map to a higher resolution for pixel classification. See Figure 7(A). The FCN first takes the whole images as input, learns deep image features through the encoder, recovers the spatial dimension by transposed convolution layers, and predicts a pixel-wise image segmentation for the left ventricle (blue region). The first issue in training a fully convolutional network (FCN) for medical segmentation is that they are computationally expensive because of the number of FLOPs required to process an image; this is why an encoder architecture is preferred [7]. The second issue of FCN is internal covariate shift where the training kernel is encumbered by the distribution of change of input features [79] and also results in unsatisfactory learning speed. In addition, a regular FCN does not provide a well-defined segmentation, which means there is no defined boundary between the pictures, and this causes ESV and EDV measurements to be inaccurate [7].

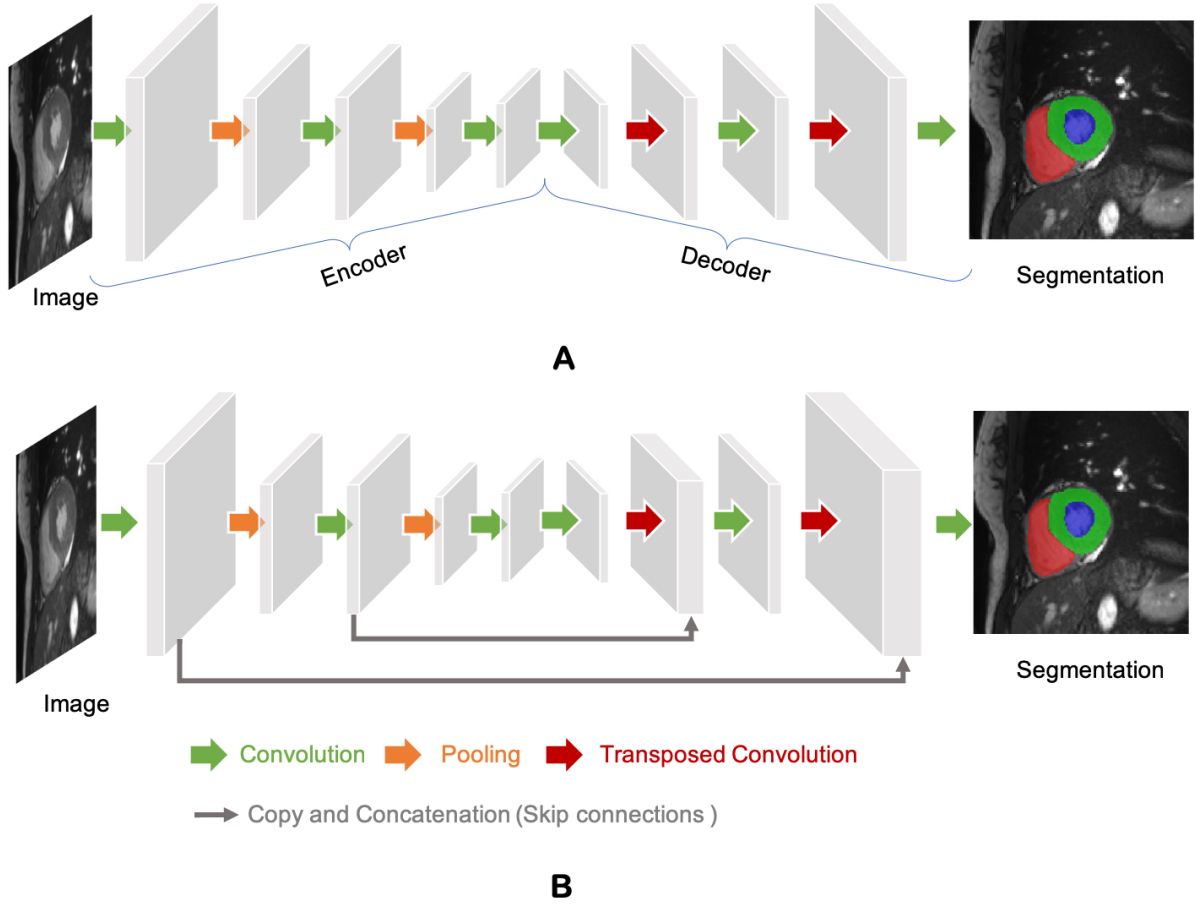


Figure 7: (A) Generic architecture of fully convolutional neural networks (FCN) for segmentation (B) A schematic drawing of U-net [113].

Several variants of FCNs have been proposed to propagate features from the encoder to decoder and to improve the segmentation accuracy. The most well-known and most popular variant of FCNs for biomedical image segmentation is the U-net [21]. See Figure 7(B).

U-net uses skip connections (the gray arrow) to carry information from the encoder to the decoder to recover spatial context loss in the down-sampling path. This approach will generate more precise segmentation.

In recent years, a large number of methods were presented using U-net to improve the performance of image segmentation. Automatic and efficient implementations have been a research focus

on the medical image computing community. Milletari et al. [66] proposed a V-shaped deformation structure called V-Net for medical image segmentation. They developed a new objective function based on the dice coefficient, enabling the network to handle situations in which there is a strong imbalance between the number of voxels in the foreground and background. Cicek et al. [67] introduced a 3D U-net model. See Figure 8.

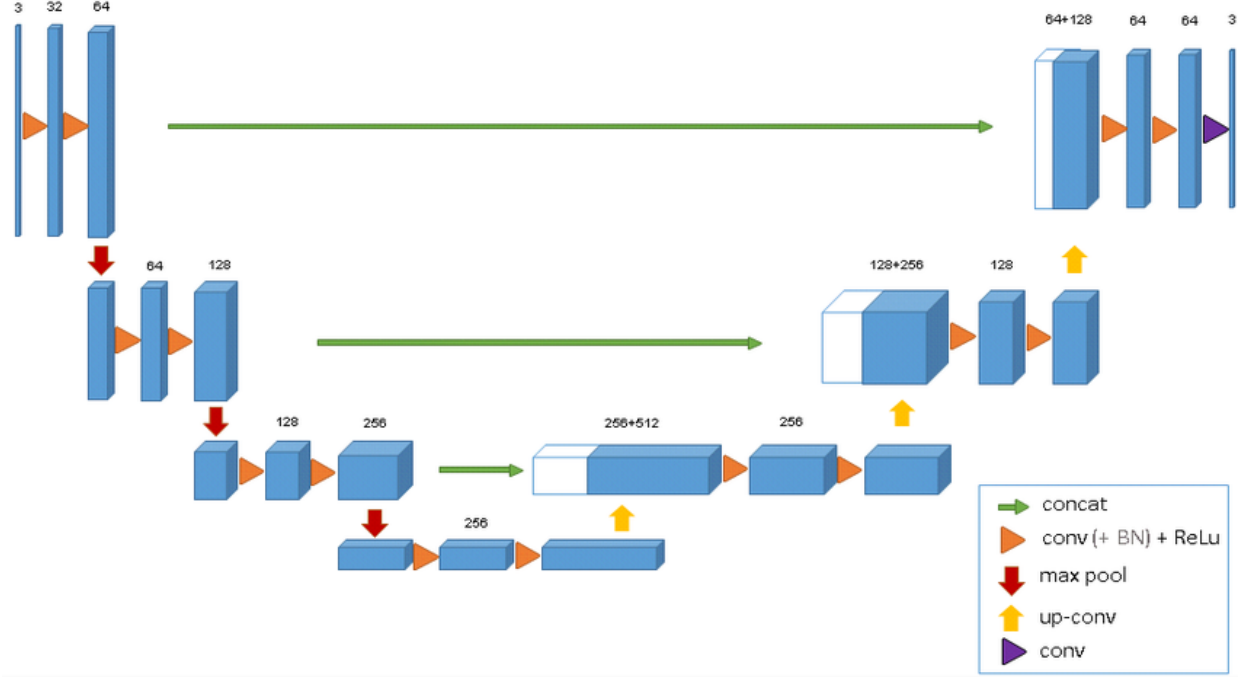


Figure 8: The 3D U-net architecture. Blue boxes represent feature maps. The number of channels is denoted above each feature map [67].

Segmenting cardiac structures in 3D is even more challenging than in 2D. The 3D U-net model is similar to the U-net that has an encoder path and decoder path. In the encoder path, each layer contains two  $3 \times 3 \times 3$  convolutions, each followed by a ReLU, and then a  $2 \times 2 \times 2$  max pooling with strides of two in each dimension. In the decoder path, each layer consists of an up-convolution of  $2 \times 2 \times 2$  by strides of two in each dimension, followed by two  $3 \times 3 \times 3$  convolutions each followed by a ReLU. There are skip connections between the encoder path and decoder path. In the last layer, a  $1 \times 1 \times 1$  convolution reduces the number of output channels to the number of labels which is 3.

The model applies batch normalization before each ReLU activation function.

Wang et al. [68] proposed the non-local U-nets, which are equipped with flexible global aggregation blocks. The network contains an encoder-decoder path with stacked local operators to aggregate information. The experiments were performed on the 3D multimodality isointense infant cardiac MR image segmentation task. The results showed that the model achieved top performance with fewer parameters and fast computation.

Ibtehaz et al. [70] presented MultiResUNet based on probable scopes for improvement U-net model architecture. The results showed that they obtained a relative improvement compared with the original U-net.

Oktay et al. [133] proposed an anatomically constrained network, which was motivated by the shape and location constraint in medical images. They demonstrated a training strategy that incorporates anatomical prior knowledge into CNNs through a new regularization model. The results showed that the proposed approach improved the prediction accuracy of the state-of-the-art models. Dong et al. [134] demonstrated that 3D echocardiography can provide full volume information of the heart and proposed a novel real-time framework for 3D LV segmentation on 3D echocardiography.

### **3.1.2 The Structure of Network**

Segmentation of medical images plays a crucial role in enabling technology for various medical applications such as diagnostics, planning, and guidance. A traditional learning process [27] faces numerous issues that limit the efficacy of the automatic diagnosis of the images.

In this research, we present the feasibility of deep learning approaches for a convolutional neural network architecture which we call BNU-net. This yields a U-shape architecture shown in Figure 9.

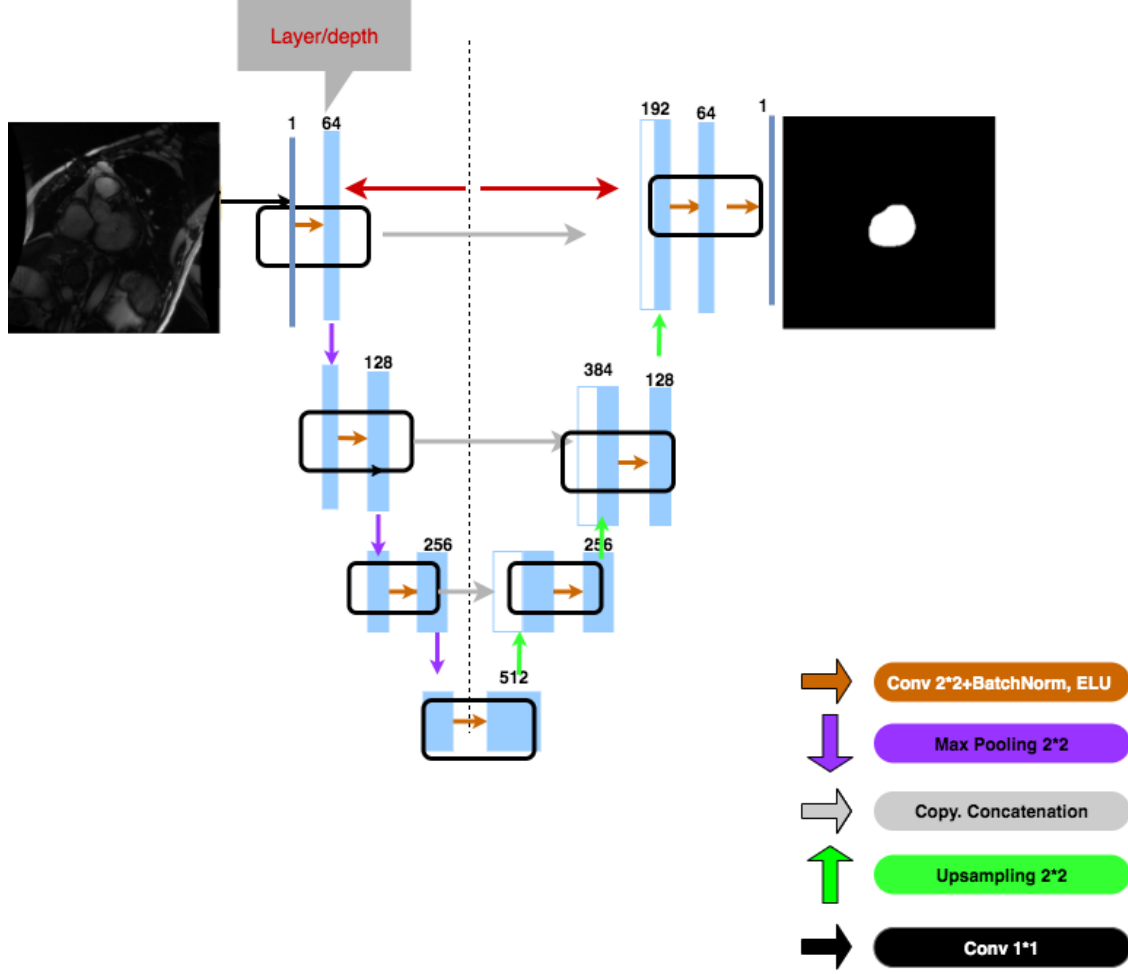


Figure 9: Architecture of the BNU-net convolutional network. (a) The contraction path is responsible for feature extraction. (b) Batch normalization is performed after each convolution in the convolutional layer.

We can see the BNU-net as being divided into two separate paths, an encoder path, and a decoder path. The encoder path reduces the spatial dimensions of each layer but increases the number of feature maps. On the other hand, the decoder increases the spatial dimensions for



each feature map while reducing the number of feature maps in each layer. We compare this approach with a parallel approach named U-net. Both BNU-net and U-net are cardiac segmentation approaches: while BNU-net employs batch normalization to the results of each convolutional layer and applies an exponential linear unit (ELU) approach that operates as activation function, U-net does not apply batch normalization and is based on Rectified Linear Units (ReLU). The presented work (i) facilitates various image preprocessing techniques, which includes affine transformations and elastic deformations, and (ii) segments the preprocessed images using the new deep learning architecture.

BNU-net has 11 convolution layers, 4 layers on the contraction path, and 7 layers on the expansion path. In the contractive path, each convolutional block corresponds to 2 convolutional layers followed by  $2 \times 2$  max-pooling layers. One of the key features of this architecture is the use of concatenation path connections between down-sampling and up-sampling layers to fuse local and global information. The expanding path is followed by a series of convolutional filters and concatenations of feature maps from the contracting path at each stage for more precise localization as shown by the gray arrows in the figure. As in the U-net architecture, the network consists of two paths: the contracting path and the expansive path. There are two main advantages of the U-net compared with other networks. Firstly, U-net needs a few training sample datasets in comparison to other existing deep learning-based methods. Secondly, different from FCN, U-net learns segmentation in an end-to-end setting which means input image is given in one end and the output segmentation is produced in the other end [76]. In the U-net model, it has a contracting and a symmetric expanding path: for each layer in the encoder network, there is a corresponding decoder layer. To recover the spatial information lost in the downsampling path and reconstruct the original size of activations, upsampling paths are employed in the decoder network. The contracting path is used to capture feature context and the expanding path is used to increase the low-resolution feature maps from the contracting path. In the U-net model, a large number of feature channels are applied in the expanding path to allow the network to propagate context information to higher resolution layers. There are six repeated applications of  $3 \times 3$  convolutions, each followed by

a rectified linear (ReLU) activation function in the contracting path. The above architecture is different from the U-net-based approach that contains four layers with batch normalization [26] and exponential linear in the contracting path.

### 3.1.3 Using Batch Normalization in BNU-net

Machine learning models need to train well on new unknown inputs, which is known as the ability of the learned model. Several regularization strategies are used in machine learning models. Two regularizations are widely used in machine learning models.

**Dropout** One of the biggest problems for deep learning is over-fitting. Because few training samples are available compared with the number of learnable parameters, many techniques have been developed to solve this problem. The term “dropout” refers to randomly drop units (along with their connections) from the neural network during training in Figure 10.

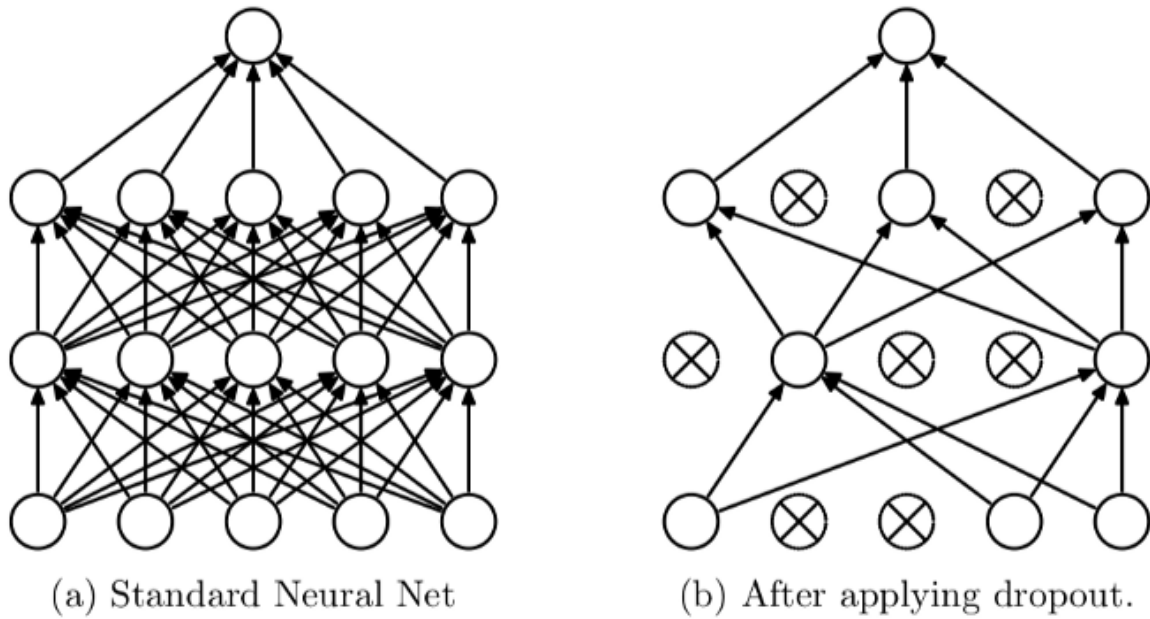


Figure 10: Dropout neural net model. Left: A standard neural net with 2 hidden layers. Right: Applying dropout to the network on the left that generates a thinned net [109].

Deep neural networks with a large number of parameters are very powerful machine learning systems [109]. It is a specific technique to deal with the overfitting problem of neural networks and to select neurons randomly that are ignored. The key idea is, during training, different thinned networks are sampled from the original model by removing a random fraction of nodes. This prevents units from co-adapting too much. Essentially, when using the dropout technique, several different classifiers with different numbers of parameters are trained separately, and then the final prediction generated by averaging all response of these classifiers are used in the test stage. At test time, it is efficient to approximate the ‘thinned’ network with smaller weights. This approach reduced overfitting and improved the performance of neural networks with state-of-the-art results.

**Batch Normalization** Batch normalization is a technique to normalize the input training data. It gives a way of accelerating training and helping to solve the internal covariate shift problem. Because the training process is affected by the parameters of each input layer, small changes of network parameters will affect the network greatly. Therefore, we use *batch normalization* [26] throughout the BNU-net, which helps bypass sharp local minima and correct activations to be zero mean and of unit standard deviation. The batch normalization is a mechanism that aims to make the training of neural networks more stable to a given network layer. We applied batch normalization after each convolutional layer and before the activation function. One of the motivations for the development of batch normalization was to allow each layer of the network to train more efficiently. It reduced problems when the input values changed and reduced internal covariate shift. We saw that when the inputs of the network were transformed such that the first two moments (mean and variance) were respectively set to zero and unit values, the network converges faster and makes the optimization landscape smoother.

### 3.1.4 Applying Activation Function

In machine learning models, the activation function decides whether a neuron should be activated or not, which serves as a gate controlling whether or not to pass the signal to the next layer.

Each neuron receives input signals and produces output signals. The gate applies a nonlinear transformation on the input signal, then feeds it to the next layer of neurons as input. Because of the non-linear transformation, the activation function is thus capable of performing more complex questions. Theoretically, the overall network can learn and show any arbitrary complex functions by imposing the non-linear transformation. Additionally, differentiability is another prerequisite for the activation function used in deep learning models, to optimize the models via the gradient descent method. Figure 11 shows several non-linear activation functions that were widely used in recent years.

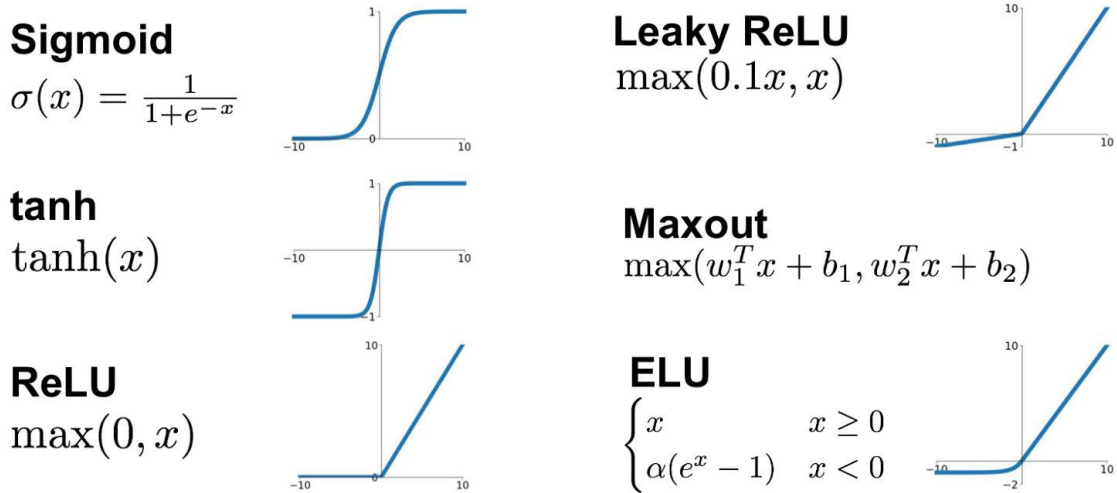


Figure 11: Activation function [108].

**Sigmoid** Sigmoid activation function, also known as logistic function, is one of the most widely used activation functions. The sigmoid function is often used in the output layer for models that predict a class probability whose range is located between 0 and 1. The main problem when using the sigmoid function is that it is affected by the vanishing gradient problem, and therefore are rarely used in the middle hidden layers. The vanishing/exploding gradient means if the network parameters are too small or too large, the gradient of the activation function would have very small values, which makes the gradients to be zero (“vanish”), and the parameters of the layers would

never be updated. The gradient of the activation function would be a large number if the input to the activation function is around zero, which will result in a big update on the learned parameters of the model.

**Tanh** Tanh is also known as hyperbolic tangent function, which is an S-shape activation function, and it performs slightly better than the sigmoid activation function. We can use tanh when performing a classification between two classes problems. The tanh will map the negative input to negative strongly and map the zero input to nearly zero.

**ReLU** ReLU is the most widely used activation function, especially in CNNs, which is defined as :  $f(x) = \max(0, x)$ . This means the output of the ReLU  $f(x)$  will be zero when the input signal  $x$  is smaller than or equal to zero. It will reduce the learning ability of the network data, since the output of the activation is always zero when  $x$  is negative, and these neurons become “dead”.

**ELU** ELU is also known as the exponential linear unit function, which is similar to the ReLU. ELU does not have the vanishing gradient or dying neurons problems. It has proved to be better than ReLU, take a lower training time, and have higher accuracy.

Different from the original U-net which used ReLU [17], we integrated exponential linear units to make the mean activations closer to zero, which helps improve the efficiency of the data compiling and calculations process [18]. we applied ReLU and ELU separately in BNU-net and the segmentation evaluation metrics as shown in Table 1. The improved method also incorporates *cropping 2D*, which is a cropping layer used to crop feature maps and concatenate multiple feature maps from the contraction path. Cropping is essential as each convolution will blur the pixels of the border of the MRI image, impacting the accuracy of the estimation.

Table 1: Comparison between our proposed BNU-net model with ELU and BNU-net with ReLU

|             | BNU-net with ELU | BNU-net with ReLU |
|-------------|------------------|-------------------|
| Dice mean   | <b>0.92</b>      | 0.90              |
| Dice std    | <b>0.04</b>      | 0.06              |
| Sensitivity | <b>0.96</b>      | 0.95              |

**Pooling** Pooling operations is to summarize statistics of subareas of previous feature maps. It reduces the spatial size of feature maps significantly and aims to achieve the feature maps’ representation more robust to small transitions and distortions. The max-pooling is considered to be the main operation of any CNN. It is usually placing pooling layers between two convolutional layers and slicing a window across the input. Pooling feeds the content of the window to calculate the local statistics, i.e. maximum or mean.

### 3.1.5 Dataset

Cardiac CINE magnetic resonance imaging is the gold-standard for the assessment of cardiac function [153]. We evaluated our approach using the dataset from the MICCAI 2009 challenge on automated left ventricle segmentation; it was downloaded from the Imaging Research Centre for Cardiovascular Interventions at Sunnybrook Health Sciences Centre [11]. The Sunnybrook dataset comprises cine MRI from 45 patients (a total of 805 images) suffering from different cardiac conditions: heart failure with infarction (12 cases), heart failure without infarction (12 cases), hypertrophy (12 cases), and healthy patients (9 cases). Each time series consists of 6 to 12 2D cine stacks with 8 mm slice thickness and 1.3 mm to 1.4 mm in-plane resolution [2]. For the purpose of the research, the MRI image files were originally split into training, validation, and testing sets in the ratio of 15 : 15 : 15. Each patient has 12 to 28 images. The database also provides MRI ground truth medical images manually segmented by an expert.

### 3.1.6 Experimental Studies

The normal unsupervised learning process [29] requires millions of parameters. Thereby, to ensure good performance, a large amount of the training data is necessary to yield a satisfactory classification. However, this is impractical given the limited amount of data available, especially in medical image computing. We present a neural network and training approach which depends heavily on information increase for more efficient use of the dataset noted. Thus, in the neural network field, one of the most widely used methods to solve this problem of limited diversity and limited quantity data is the artificial increase of training data.

**Data augmentation** To ensure good performance as well as generalization of our mode, a large training dataset that is diverse enough to represent the overall data distribution is necessary. However, in medical image computing, especially in cardiac medical segmentation, few datasets are available for researchers. Thus, one of the most widely used methods to overcome the problem of limited quantity and limited diversity of data is to generate a new training dataset by augmenting the original training dataset. There are two ways to generate more images. One way is to change original images with geometry transformation, such as flip, rotation, and translation. The other way is to use generative models, especially Generative Adversarial Networks (GANs) [98]. GANs can create synthetic images and add them to the training set. There are some applications to prove that adding these synthesized images into training can help the training procedure [99].

Specifically, the improved methods of processing use elastic deformations and random affine transformations to do data augmentation. Elastic transform was first proposed by Simard et al. [30] in 2003. It was first applied in the MNIST handwritten digit recognition data set. The implementation performs on-the-fly elastic transformations for efficient data augmentation during training. We also perform an affine transformation (rotation, scaling, translation) to augment the training set, mitigate overfitting, and improve generalization. Figure 12 shows examples of using elastic transform and random affine transformation.

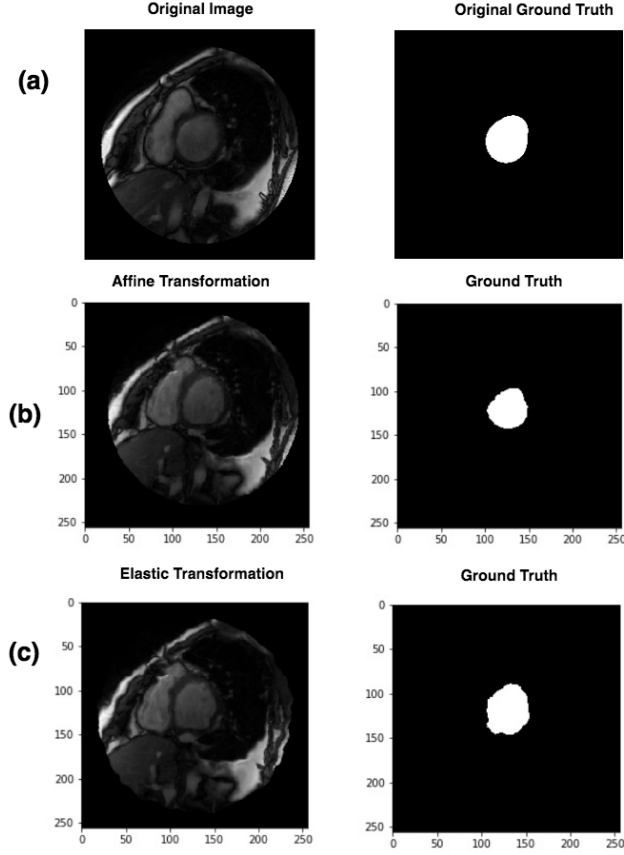


Figure 12: The contour comparison based on MRI input. (a) Sunnybrook Image and corresponding label. (b) Perform affine transformations. (c) Perform elastic transformation to augment the training set.

We applied elastic deformations to the available training images in our data set. The elastic deformation enables the neural network to grasp invariance to this kind of deformation without seeing these changes throughout the annotated corpus of the picture. Such a process is especially essential in biomedical segmentation as deformation is one of the most prevalent tissue variations, which allows effective simulation of realistic deformations.



### 3.1.7 Evaluation Metrics

To measure the performance of the BNU-net, we provide an overview of the main metrics reported in the literature for comparative purposes.

Let  $A$  and  $G$  be the automatically segmented and ground truth (manual) region/contours, respectively.

1) The dice metric: a measure of contour overlap between automatically and manually segmentation, and is defined as:

$$Dice = \frac{2(A \cap G)}{A + G} \quad (1)$$

where  $A \cup G$  denotes the intersection between  $A$  and  $G$ , and  $A \cap G$  means the union between  $A$  and  $G$ . The dice index varies from zero to one. Zero indicates a total mismatch with the ground truth and one indicates there is a perfect match.

2) Average perpendicular distance (APD) measures the distance in mm between contours and averaged over all contour points. A low value implies that the two contours match closely. APD is computed in millimeters with spatial resolution obtained from the Pixel Spacing DICOM field [2].

3) In addition to the two similarity measures above, two additional metrics used to evaluate the performance of the segmentation are sensitivity and specificity. They are computed using the formulas:

$$Sensitivity = \frac{T_0}{B_0} \quad (2)$$

$$Specificity = \frac{T_1}{B_1} \quad (3)$$

where  $T_0$  and  $T_1$  are the total numbers of correctly predicted object and background pixels, respectively. The total number of object and background pixels are denoted by  $B_0$  and  $B_1$ , respectively.

### 3.1.8 Segmentation Results and Comparison

This section provides examples of segmentation results with good accuracy produced by the proposed BNU-net architecture on the test set. Table 2 summarizes the results of BNU-net with and without data augmentation and compares them to U-net results.

Table 2: Output of the model and efficiency metrics results

|             | U-net with data augmentation | U-net without data augmentation | BNU-net with data augmentation | BNU-net without data augmentation |
|-------------|------------------------------|---------------------------------|--------------------------------|-----------------------------------|
| Dice mean   | 0.88                         | 0.87                            | <b>0.93</b>                    | 0.92                              |
| Dice std    | 0.09                         | 0.11                            | <b>0.03</b>                    | 0.04                              |
| Sensitivity | 0.96                         | 0.95                            | <b>0.97</b>                    | 0.96                              |

From the graph, we can see that with and/or without data augmentation, the BNU-net achieved the best dice score and sensitivity score. We found that the left ventricle recognition is improved when using data augmentation.

This fast training time with limited resources, including the amount of the images and the processing power, has made it possible to apply this methodology to more settings. Among the methods in the Sunnybrook dataset, four groups report their performances on the 45 patient cases that have the same data and split. The performances on these cases of BNU-net and these methods are shown in Table 3.

Table 3: Segmentation results on the Sunnybrook dataset, compared to the performance from the state-of-the-art methods

|                                   | Dice Mean   | Dice Std    | APD (mm)    |
|-----------------------------------|-------------|-------------|-------------|
| BNU-net with data augmentation    | <b>0.93</b> | <b>0.03</b> | <b>1.94</b> |
| BNU-net without data augmentation | 0.91        | 0.04        | 2.06        |
| Huang et al. [6]                  | 0.89        | 0.04        | 2.16        |
| Ngo and Carneiro [3]              | 0.90        | <b>0.03</b> | 2.08        |
| Hu et al. [5]                     | 0.89        | <b>0.03</b> | 2.24        |
| Liu et al. [8]                    | 0.88        | <b>0.03</b> | 2.36        |

We find that BNU-net with data augmentation achieves the best dice mean, dice std, and APD performance compared with other groups' results.

Figure 13 shows two examples of the output segmentation. We have demonstrated the efficacy and utility of the BNU-net architecture for semantic segmentation in cardiac MRI.

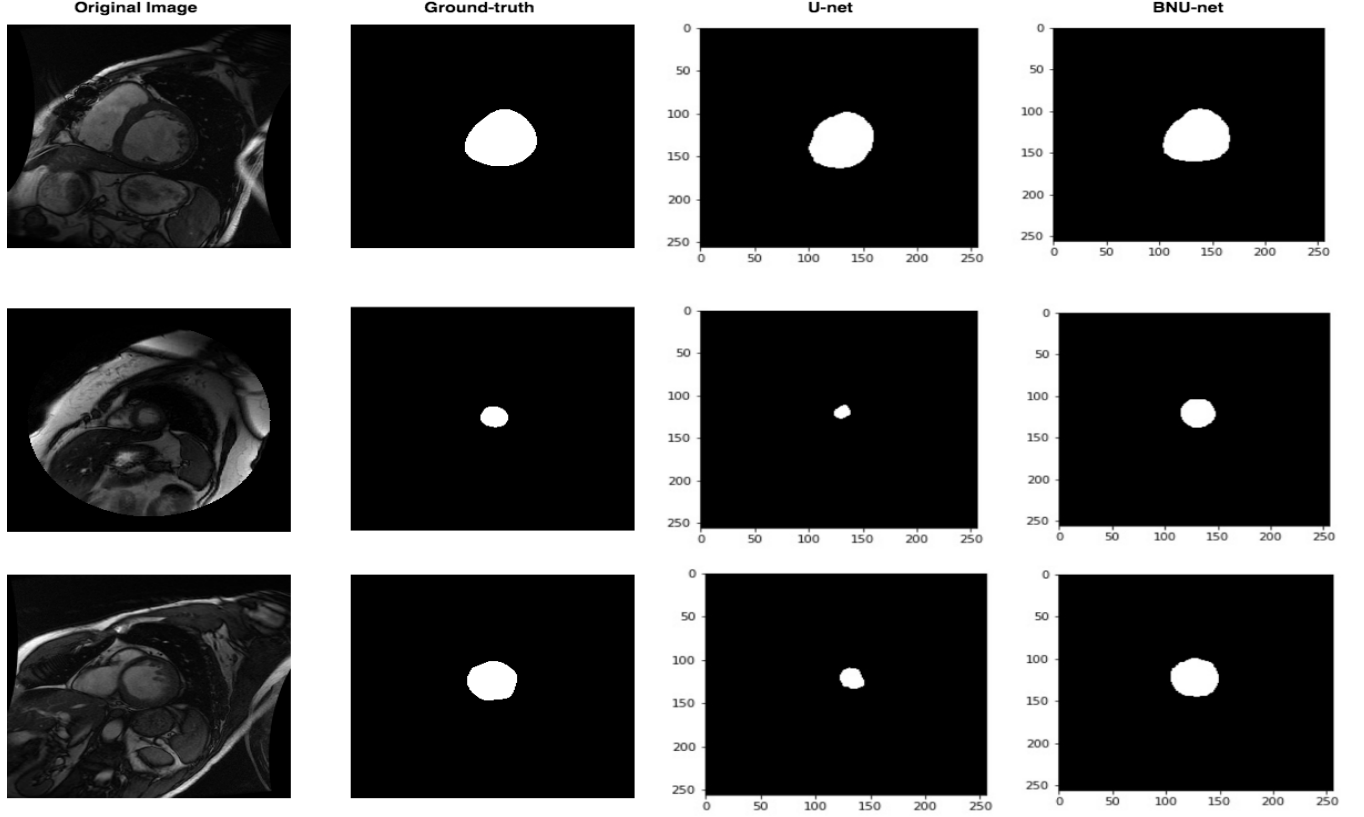


Figure 13: Segmentation predictions on raw inputs from the Sunnybrook dataset.

The quantitative evaluation of the BNU-net is based on the dice coefficient and the average perpendicular distance.

### 3.1.9 Discussion

Medical image segmentation is one of the most important preliminary steps in the identification of the tissues in image scans. Deep learning has become one of the most important powerful tools in solving computer vision tasks including segmentation. U-net was originally proposed for general biomedical image segmentation [21] and was widely accepted among researchers for image segmentation tasks. In this research, we proposed a novel method called the BNU-net model for the segmentation of the left ventricle. Compared with the original U-net model, the proposed BNU-net model is able to re-establish contextual information and lead to more accurate segmentation

results. The BNU-net model consists of two main paths which are the encoding path and the decoding (up-sampling) path. In the encoding path, successive layers consist of the convolutional filters followed by an activation function (ELU) which is used to learn a representation of the input image. To solve the internal covariate shift and help the network train faster and achieve higher accuracy, one important modification in the model is that we applied batch normalization after the convolutional filters. We use batch normalization throughout the BNU-net, which helps bypass sharp local minima and correct activations to be zero-mean and of unit standard deviation. It allows us to prevent small changes in layer parameters from amplifying through the deep network [21]. In addition, we are less careful about initialization. A series of  $2 \times 2$  max-pooling layers were used to down-sample the output and capture the features crossing different layers. The decode path consists of the up-sampling of the channels followed by convolutional blocks and ELU. One of the important steps in up-sampling is the concatenated connection from the encode path, which is used to propagate context information to detect the fine, higher resolution features.

We applied data augmentation when training our model by applying affine transformations and elastic deformations to the Sunnybrook biomedical segmentation dataset. It achieves outstanding performance lines in fast-growing GPU computability, only needs a few annotated images, and achieves high training speed on each epoch (11 seconds) with a batch size of 16 on an NVIDIA GeForce Titan X Pascal GPU. At test time, our model segments the whole series of images in less than 12 seconds. We also used the smaller images size of  $176 \times 176$  pixels, which yielded slightly better performance results on the testing set and faster training time. With 1 GPU and 200 epochs,  $256 \times 256$  images took 36 minutes to complete training, while  $176 \times 176$  images took 30 minutes.

## **3.2 Two Deep Learning Approaches for Automated Segmentation of Left Ventricle in Cine Cardiac MRI**

### **3.2.1 Methodology**

U-net is a convolutional neural network (CNN) specially designed for biomedical image segmentation [21]. U-net network is composed of U-shape and skip-connection, which is efficient in using

GPU memory. Because GPU memory is a bottleneck compared to its computation power, saving GPU memory is quite a significant advantage for CNN. U-net also outperformed related work because it required fewer annotated training samples. As we know, it is quite expensive to acquire annotated samples, so U-net is excellent in getting high accuracy with less training data. There were some variants based on U-net, including UNet++ [80], Attention UNet [69] and BNU-net [31].

Zhou et al. [80] proposed UNet++, which is a new and more powerful architecture for medical image segmentation. Their network is composed of encoder and decoder networks, where encoder and decoder are connected through skip pathways. In U-net, the skip connections structure links the high-resolution encoder with decoder output, which results in the concatenation of semantically dissimilar feature maps. They re-designed the skip connection aimed at reducing the semantic gap between the feature maps of the encoder and decoder networks. The experimental results demonstrated that UNet++ can capture fine-grained details of 2D images.

Okta et al. [69] proposed a novel attention gate (AG) model for automatic medical image segmentation. It learned to focus on target structures of varying sizes and shapes. AGs could learn to highlight the useful features for images while ignoring the irrelevant areas. Experimental results showed that AGs improved the prediction performance of U-net across different datasets.

We use three normalization methods in this study, which are batch normalization [26], layer normalization [32], and instance normalization [33]. They can generally be viewed as a technique for data preprocessing. For the improvement of accuracy, it is especially effective when features are widely distributed because all of the features are forced into a similar range. It also helps to reduce the internal covariate shift, which means it weakens the changes in network parameters during training. The larger the difference in feature range, the more effectively the data normalization method will restrict the weights into a certain range and prevent them from exploding. Therefore, the optimization process will become faster.

Batch normalization [26] is the first method we test. It normalizes the output of previous activations across the batch in a network. You can view it as doing data pre-processing at each layer of the neural network. When we use batch normalization, we insert a batch norm layer into

the U-net. See Figure 9.

Batch normalization has a reputation for speeding up the training as well as the generalization of convolution neural networks(CNN) [26]. However, for recurrent architectures, this technique fails greatly since its application is only limited to stacks of RNN, where the application of normalization is vertical to avoid the problem of repeated rescaling.

Batch normalization harbors a few weaknesses that limit its ability to be considered the most influential technique to reduce the internal covariant shift (ICS) in deep neural network models [26]. During the normalization of outputs from previous models, the batch is divided by the empirical standard deviation and the result is subtracted from the subsequent empirical mean. Batch normalization is typically poor in the pipelining of online learning, thereby contributing to poor generalization of the training data that is contributed by the change in batch size. This resulting shift in the input data eventually affects its performance. To solve this problem, other normalization methods were proposed.

Layer normalization [32] normalizes the input across features, which is completely different from normalizing across batches in batch normalization. The calculation for mean and standard deviation is similar to that in batch normalization. The mean across the features and variance across the features are calculated for each element of the input. We applied layer normalization in each convolutional block and based it on an exponential linear unit (ELU). See Figure 14.

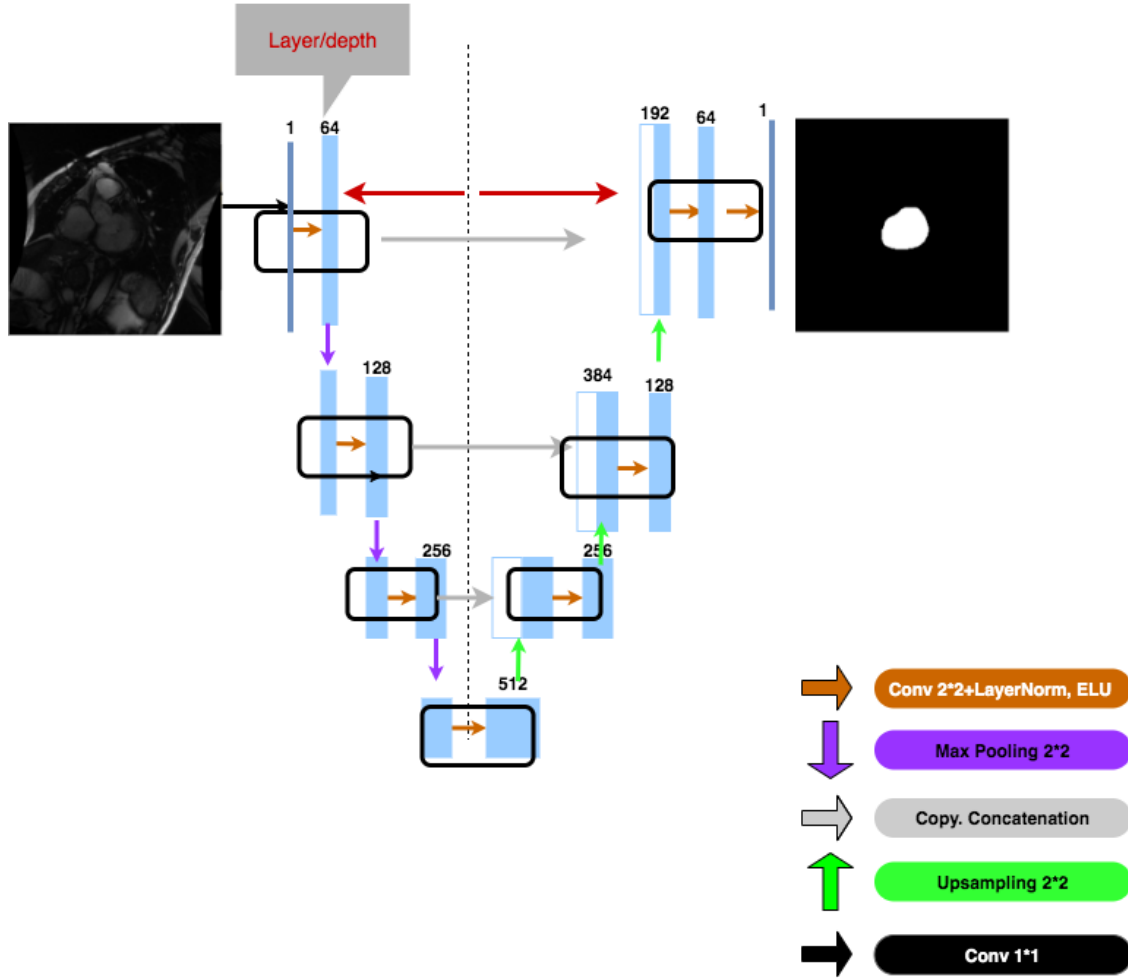


Figure 14: The LNU-net architecture of the proposed fully convolutional network. Layer normalization is performed after each convolution in the convolutional layer.

In batch normalization, the statistics are the same for each batch, while in layer normalization, the statistics are computed across each feature and are independent of other examples. The independence between examples makes it simple when applied to recurrent neural networks. The



authors of “Recurrent Fully Convolutional Neural Networks for Multi-slice MRI Cardiac Segmentation” [1] found it outperformed other normalization methods in the application of RNNs.

Batch instance normalization [34] is an interpolation of batch normalization and instance normalization. With instance normalization, the mean and standard deviation are computed across each channel for each input group. Instance normalization shows some similarity with layer normalization, but there is one obvious difference: layer normalization calculates the statistics across each feature in the training sample, while instance normalization calculates across each channel. In recent years, instance normalization was also found to be effective for Generative Adversarial Networks (GANs). However, there are also problems with instance normalization: it is suitable for style transfer cases but incapable of contrast matters. This problem makes it unsuitable for image segmentation tasks. To solve this problem, we use instance and batch normalization together in the first convolutional block, apply an ELU and pass its result to the next layer. See Figure 15. It balances the batch normalization and instance normalization, and the model could learn to use different combination percentages with gradient descent. According to the previous work [34], batch instance normalization outperforms batch normalization on CIFAR-10/100, ImageNet, adaptation, style transfer, and image classification tasks.

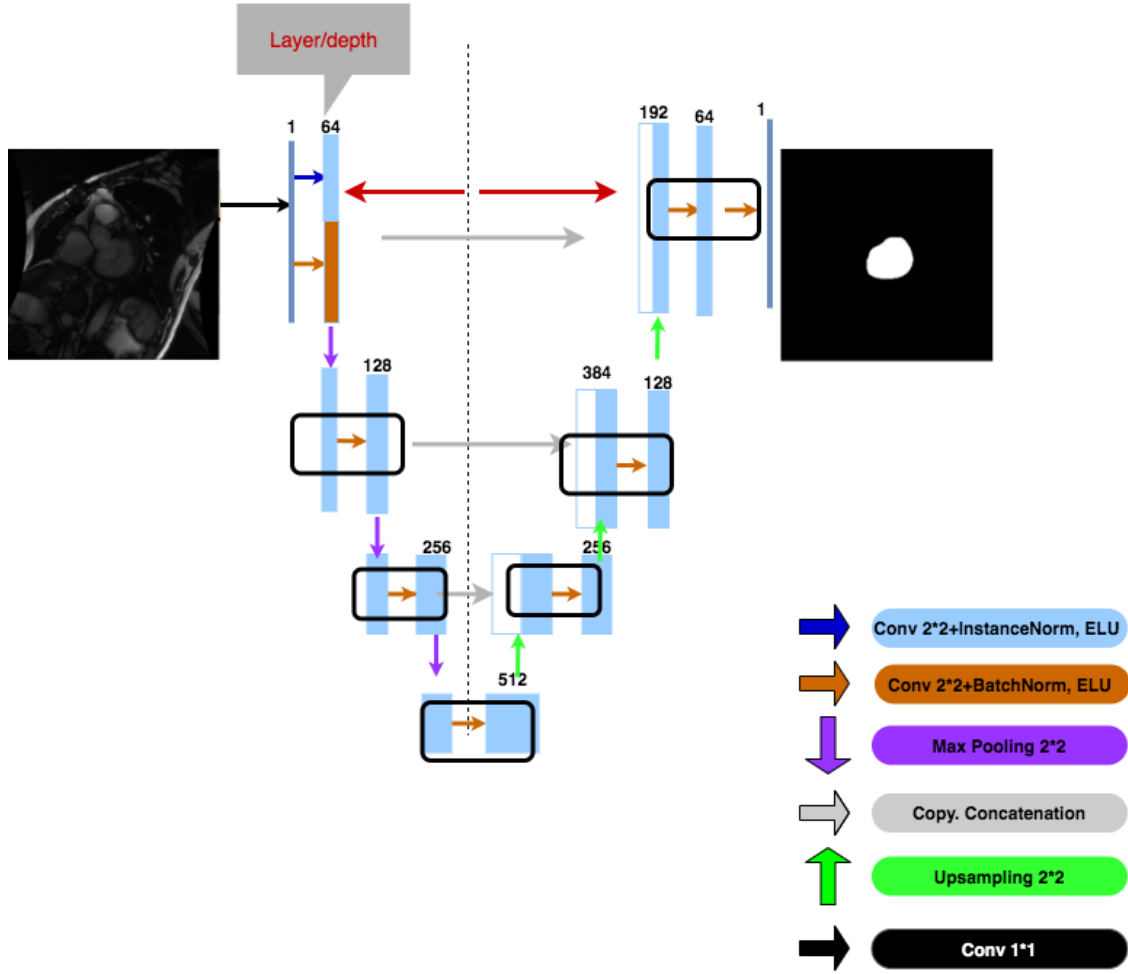


Figure 15: Architecture of the IBU-net convolutional network. Instance normalization is applied in the first convolutional layer. Batch normalization is performed after each convolution in the convolutional layer.

### 3.2.2 Dataset

The Sunnybrook dataset [35] was used for the experiments. It was received from Imaging Research Centre for Cardiovascular Intervention in Sunnybrook Health Sciences Centre and made for automatic left ventricle segmentation for MICCAI 2009. The ground truth masks were provided by experts that were manually segmented. The experiment was conducted with an NVIDIA GeForce Titan X Pascal GPU. When we compare the computational speed for each epoch, the performance of computational resources should also be considered.

### 3.2.3 Model

In this study, the model we used for image segmentation was a modified U-net based on a fully convolutional network. We added Exponential Linear Units (ELU) as an activation function instead of ReLU in the original version. In addition, we also added a normalization layer into the model, which was not considered in the original U-net. Refer to Figure 9, Figure 14, and Figure 15, for the architecture of models.

- Conv2\*2 is a convolution layer with a 2\*2 kernel.
- ELU is applied instead of ReLU to speed up the training process. It also avoids the vanishing gradient problem, because the value is no longer 0 with a negative  $x$ .
- Normalization: Normalized feature map to solve internal covariate shift.
- DownSampling layer is used to extract and interpret the contextual information on the input image.
- Cropping2D is a cropping layer designed for cropping feature maps. It is also used to reduce concatenating to avoid overfitting.
- Concatenating is two connecting different feature maps from downsampling.
- UpSampling layer is used to perform on the decoding path and increase the size of the feature which has the same size as the input data.

As we see, the new U-net for automated left ventricle segmentation is based on the basic structure of a fully convolutional neural (FCN) network architecture [79]. The first improvement

we made was an encoder architecture. It is a computationally expensive process to train an FCN for medical image segmentation, because of the large number of multiply-and-adds required for image processing. An encoder architecture would help to ease the computational burden and quicken the training process, especially when the computational power of Titan X was limited.

The second issue is the improvement of normalization. As we mentioned, three normalization methods are attempted for model improvement, which are batch normalization, layer normalization, and batch instance normalization. We found that for medical image segmentation, internal covariate shift is a problem during training with an FCN. This was encumbered by the distribution of input feature change [79]. It also leads to a slow down in training. Because training was a process independent of initial data distribution, we thought it was a good attempt to involve data normalization in our study.

The third issue is to skip connections in FCN. We found it lacked well-defined edges with FCN segmentation. The boundaries between the images were not clearly defined with the original version of FCN. The skip of connection was supposed to be a helpful in accuracy.

### **3.2.4 Experimental Studies**

In order to achieve great learning performance, we need a large number of the labeled dataset. However, in current medical areas, there are limited training samples. Therefore, the data augmentation methods are applied in medical image datasets. Patrice et al. [30] first proposed the elastic deformation in 2003. We use elastic deformation to the available training images that increase the size of the training dataset and improve the adaptability of the models. In this work, we use the following data augmentation strategies: affine transformation, elastic deformation, and rotation.

The original version of U-net is a typical method for left ventricle cardiac image, so we first test its performance with an NVIDIA GeForceTitan X Pascal GPU. We found that the dice mean was only 0.87 with U-net. It took 11s per epoch on training if we set the batch size as 16. In this study, a novel segmentation schema that contains batch normalization and instance normalization was proposed to improve the segmentation performance. We want to improve the dice mean and

to accelerate the training process.

### 3.2.5 Segmentation Evaluation

We use U-net with batch normalization, U-net with layer normalization, and U-net with batch-instance normalization to deal with image segmentation tasks. The dice mean is listed in Table 4.

Table 4: Dice mean with different normalization

| Dice Mean                    | With ELU    | With ReLU |
|------------------------------|-------------|-----------|
| Instance-Batch Normalization | <b>0.94</b> | 0.93      |
| Batch Normalization          | <b>0.91</b> | 0.90      |
| Layer Normalization          | <b>0.89</b> | 0.88      |

ELU was likely to introduce more calculations compared to ReLU. But when we had a negative  $x$ , ELU would avoid the gradient vanishing led by ReLU. In addition, ELU almost shared all benefits with ReLU. Table 5 displays the dice mean for six experiments: batch normalization with ELU and ReLU, layer normalization with ELU and ReLU, as well as batch instance normalization with ELU and ReLU. We would like to test the influence of different activation functions on image segmentation, on both dice mean and training speed. In the meanwhile, we also wanted to know whether normalization methods would lead to better performance, and which normalization was optimal in this case.

Table 5: Output of the models and efficiency metrics results

|                                        | Dice Mean   | Dice Std    | Sensitivity | APD (mm)    |
|----------------------------------------|-------------|-------------|-------------|-------------|
| IBU-net with data augmentation         | <b>0.96</b> | <b>0.02</b> | <b>0.98</b> | <b>1.91</b> |
| IBU-net without data augmentation      | 0.94        | 0.03        | 0.96        | 2.02        |
| LNU-net with data augmentation         | 0.90        | 0.11        | 0.96        | 2.29        |
| LNU-net without data augmentation      | 0.89        | 0.14        | 0.95        | 2.46        |
| BNU-net [31] with data augmentation    | 0.93        | 0.03        | 0.97        | 1.94        |
| BNU-net [31] without data augmentation | 0.91        | 0.04        | 0.96        | 2.06        |
| U-net with data augmentation           | 0.88        | 0.09        | 0.96        | 2.48        |
| U-net without data augmentation        | 0.87        | 0.11        | 0.95        | 2.51        |

The performance of LNU-net (Figure 14) and IBU-net (Figure 15) is shown in Table 5. This table illustrates how LNU-net and IBU-net vary with and without data augmentation. When comparing three normalization methods in Table 5, we found that batch instance normalization is the best choice for image segmentation. Batch instance normalization combined the benefits of batch normalization and instance normalization, getting better performance in image segmentation tasks.

Compared to the original version of U-net, the combination of ELU and batch instance normalization improved the dice mean by 8% from 0.87 to 0.96, which is a significant and reliable improvement. The improvement in training speed was not displayed in the table above. Each epoch was trained with 11 seconds in the original U-net. While with the combination of encoder architecture, batch instance normalization, and drop-connection, it only took 8 seconds to train an epoch. The training speed was improved by 27%, which meant almost one-third more images could be processed within the same period of time. The acceleration was due to the encoder architecture and drop-connection. These could reduce the calculation during training and they could also avoid overfitting.

To evaluate the performance of our proposed structures, we use the same Sunnybrook dataset

and same data split, compare the performance of our methods with other groups report their performances on the 45 patient cases. The amount of improvement in performance varies across different normalization methods. In Table 6, the observation is that using IBU-net with data augmentation achieves the best dice mean, dice std, and average perpendicular distance when compared with other groups’ proposed structures.

Table 6: Dice score and average perpendicular distance (APD) of segmenting the Sunnybrook dataset, compared to the performance from the state of the art methods

|                      | Dice Mean   | Dice Std    | APD (mm)    |
|----------------------|-------------|-------------|-------------|
| IBU-net              | <b>0.96</b> | <b>0.02</b> | <b>1.91</b> |
| X Zhou [28]          | 0.92        | -           | -           |
| Zhou [36]            | 0.93        | 0.06        | -           |
| Ngo and Carneiro [3] | 0.90        | 0.03        | 2.08        |
| Hu et al. [5]        | 0.89        | 0.03        | 2.24        |
| Huang et al. [6]     | 0.89        | 0.04        | 2.16        |
| Liu et al. [8]       | 0.88        | 0.03        | 2.36        |

Figure 16 and Figure 17 show three examples of the output segmentation suffering from different cardiac conditions: heart failure with infarction, hypertrophy, and healthy patients.

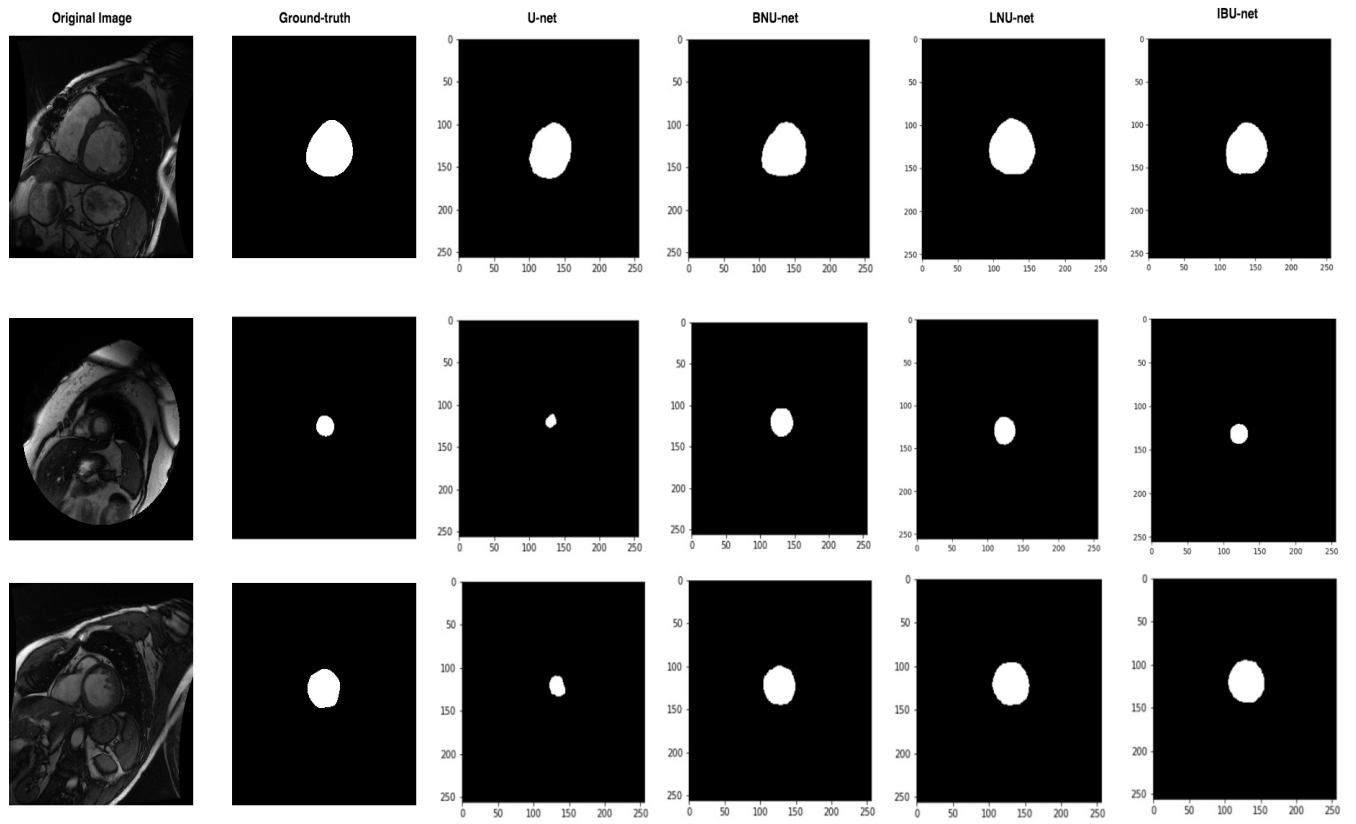


Figure 16: Examples of segmentation results on raw inputs from three conditions in the Sunnybrook dataset. The first row contains heart failure with infarction, the second row represents hypertrophy, the third row shows healthy patients.



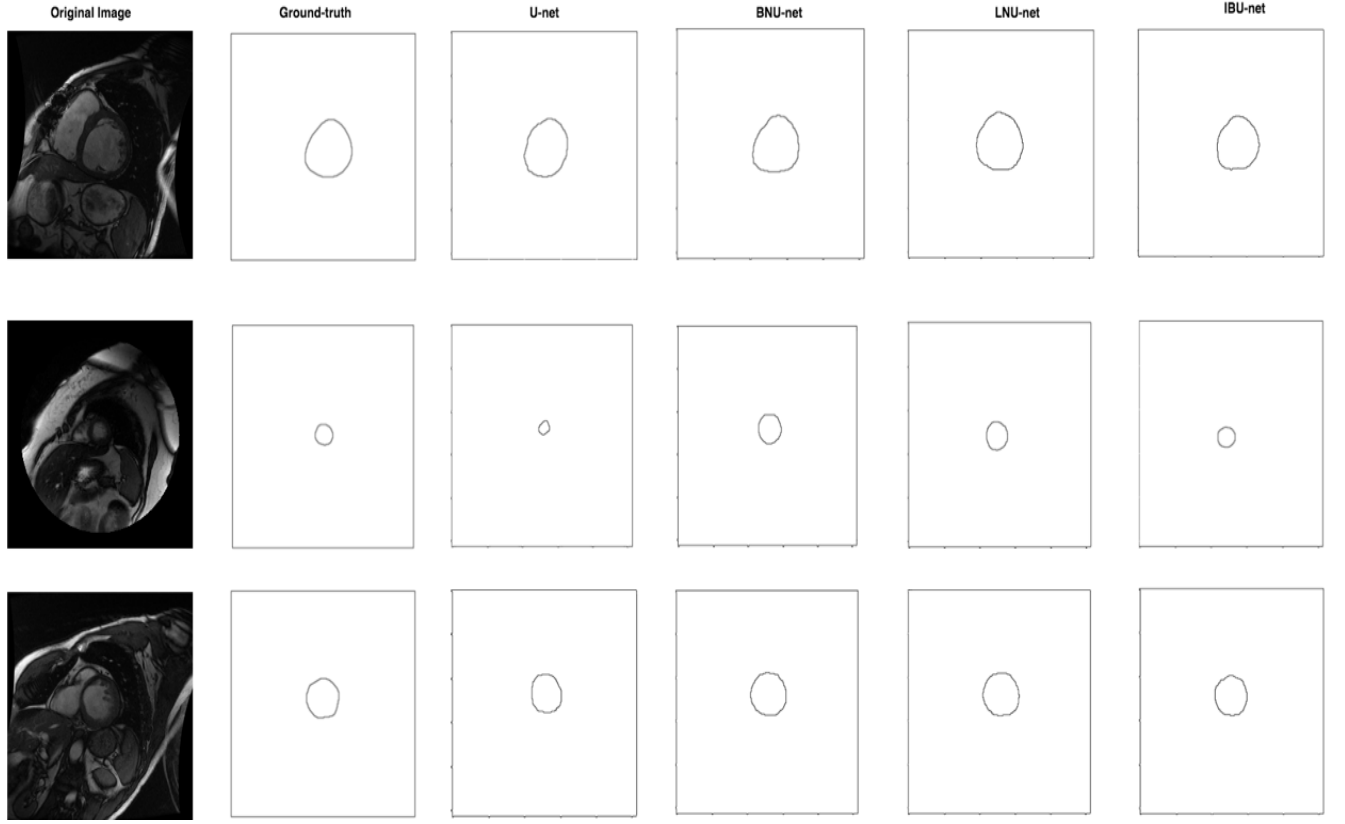


Figure 17: Some segmentation outputs by our methods. The solid lines represent the segmentation examples of the Sunnybrook dataset. We compare four different network methods, which are U-net, BNU-net, LNU-net, and IBU-net respectively.

The experimental results for the algorithm above indicate that our proposed methodology has improved efficiency and effectiveness, and outperforms convolutional deep networks contributed by the increase in the resultant dice score.

### 3.2.6 Discussion

In this study, we proposed two novel convolutional neural networks based image segmentation models. We improved the left ventricle cardiac image segmentation tasks with a renewal U-net. We have categorically explored some state-of-the-art segmentation models and a comparison among the performance of those models on the same dataset. Data preprocessing was conducted with batch

normalization, layer normalization, and batch-instance normalization. ELU took the place of ReLU as the new activation function. Encoder architecture and drop-connection were also conducted to accelerate training and reduce over-fitting.

## 4 Simulations of MRI Guided and Powered Ferric Applicators for Tetherless Delivery of Therapeutic Interventions

In this chapter, we demonstrate the effectiveness of our proposed models by simulating MRI-guided MRbots for wireless delivery of therapeutic interventions. In the following sections, we will first briefly introduce the MRI-driven device along with previous work done in the area. We then demonstrate virtual fixtures (VF) we use to track the path, including both trajectory path and MRbot movement. In addition, we illustrate the theoretical model and MRI gradients in order to fully explore the therapeutic interventions from different modalities. An experimental platform is provided, which is the QT platform. In the end, we present the simulation results.

### 4.1 Methods

#### 4.1.1 MRI-driven Devices

The goal of the present research is to simulate the movement of a sphere through 3D models of vessel corridors. Because the human blood circulatory system is made of arteries, veins, and capillaries, the respective diameters of the pathways through which a biomedical robot must pass range from tens of micrometers to several centimeters. It becomes obvious that being able to reduce the size of such an MRbot would allow a larger percentage of locations in the human body to be reached. In many cases, an untethered implementation is suitable in order to propel MRbots into the human blood circulatory system, and it is obvious that propelling such wireless MRbots in the human cardiovascular system with existing technologies is a great technical challenge [37]. A large number of methods were presented using robotic manipulators [57]-[59]. The use of an external magnet (electromagnetic fields) of the MRI can be used to directly apply forces on a tetherless object through gradient manipulation [60], by inducing a current inside the micro-coils maintaining the robots to swim to their designated target [61], or even steering a catheter to its destination [62].

#### 4.1.2 Overview the VF Approach

In order to maneuver safely, we introduce virtual fixtures (VF) which are virtual software-based constraints. There are two main parts with which we are concerned: one is continuous tracking and path imaging, the other one is maneuvering safely. Several means of propulsion are proposed to be embedded into such an MRbot [44][45].

We propose a trajectory plan, which aims to follow a planning path to ensure safety and to control the MRbots. We use VF to generate virtual guidance cues for directing MRbot inside the vessels. When we use the guidance cues, the MRbot should reach the destination point accurately and not hurt the vessels. In the proposed approach, we simulate MRI-guided and powered ferric MRbots for tetherless delivery of therapeutic interventions. The size of the ferromagnetic body must be selected according to the diameters of the blood vessels, which is then used for the predefined path to the final target location. In order to propel such MRbots into the body fluids, especially in the human blood circulatory system, it must be considered part of the normal blood flow. The smaller these MRbots are, the wider the operating range becomes through access to the blood vessels such as capillaries. The system is shown here and on which we will focus is used in the case of a cardiac tumor treatment through localized drug delivery. To ensure safety and accuracy, such procedures require continuous on-the-fly imaging (to both tracks the MRbots and image the pathway) and propulsion.

Our goal is to control and propel the micro-robot by magnetic field gradients inside the blood vessels. We use a proportional integral derivative (PID) controller to correct and stabilize the trajectory when the sphere moves in more tortuous segments of the vessel. The physical parameters of the blood flow and the diameters of the blood vessels that are targeted determine the drag force and the constraints experienced by the MRbots. The quantity of the ferromagnetic particles is performed and determined by the medical application.

## 4.2 Theoretical Model

According to the present invention, a homogeneous spherical MRbot is considered in this research. Since the gradients are generated uniformly throughout the border of the scanner, forces can be produced everywhere inside the body and can be precisely controlled with respect to time thus making it the ideal solution. In small blood vessels, the buoyancy of the MRbots and their weight can be neglected because such parameters are smaller than the drag force by orders of magnitude. The blood flow in the arterial system is pulsatile and much faster at the exit of the heart. The blood velocity decreases away from the heart to a cm/s then mm/s [38].

### 4.2.1 Interpolation

Interpolation is an important reference for medical imaging applications. In the volumetric imaging of this research, we use an interpolation method called Piecewise Cubic Hermite Polynomials (pchip) to compensate for nonhomogeneous data sampling. Given an interval  $[a, b]$ , a function  $d : [a, b] \rightarrow \mathfrak{R}$  with derivative  $d' : [a, b] \rightarrow \mathfrak{R}$ , we use a cubic Hermite spline  $s$  that approximates  $d$  over  $[a, b]$ . Our method will use the nearest method to split the interval  $[a, b]$  into  $N$  subintervals of nodes  $\vec{x} = (x_0, x_1, x_2, \dots, x_N)$  with  $a = x_0 < x_1 < x_2 < \dots < x_N = b$ . In order to calculate 3D interpolation, we introduce a new parameter  $T$  as a tuple (pathDistance,  $x$ ), where pathDistance equals the distance from the start point to the current point.

### 4.2.2 Calculating the Tortuosity

Tortuosity is defined as the minimal geometrical path in a media which may require the computation of the minimal geodesic trajectories. At this point, we indicate that tortuosity is a directional parameter. It is a parametrically-defined space curve in three dimensions given in Cartesian coordinates by  $K = (x(t), y(t), z(t))$ , corresponding to the three Cartesian axes ( $X$ ,  $Y$ , and  $Z$ ).

$$K = \frac{\sqrt{(z''y' - y''z')^2 + (x''z' - z''x')^2 + (y''x' - x''y')^2}}{(x'^2 + y'^2 + z'^2)^{\frac{3}{2}}} \quad (4)$$

where  $x', y', z'$  are the first derivative of the desired path,  $x'', y'', z''$  are the second derivative of the desired path.

#### 4.2.3 Blood Velocity

Pulsation arises from the pumping action of the heart, which forces a pulsating blood flow into the arteries, thereby creating a time-varying pressure that acts on the vessel wall [39]. The models are zero-dimensional (0D) models of the circulation, to one-dimensional (1D) models of blood pressure and flow propagation [40], to three-dimensional (3D) fluid-structure interaction techniques. Each approach has its own merits and limitations. We use 3D methods to represent complex flows, wave propagation, and blood flow–vessel wall interactions. The drag force produced by the blood acting on a sphere in an infinite extent of fluid can be calculated as:

$$F_{drag} = \frac{1}{2} \cdot C_d \cdot p \cdot Re \cdot |V_{blood} - V_s| \quad (5)$$

where  $C_d$  is the drag coefficient,  $p$  is the blood density ( $1.025 \text{ kg/m}^3$ ),  $Re$  is the reference area.

#### 4.2.4 Different Type of Blood Flow

In the present work, we studied the blood flow in the three types of pulsative: constant, normal, and high heart rate flow [41]. The gradients are updated every 100 ms but the position of the sphere can be found every  $T_p$  millisecond. We use different  $T_p$  values in the different flows to test and show the results in the Qt platform.

#### 4.2.5 Velocity of Sphere

The trajectory generates the desired velocity  $V(r)$  along with the track, counting its vector for each one of the points of the path [42][43].

$$V(r) = \frac{V_0}{1 + K/K_0} + \frac{R_s - R_{GC}}{R_0} \quad (6)$$

where  $R_s$  is the radius of the MRbot,  $V_0$ ,  $K_0$ , and  $R_0$  are constants allowing for adjustment of the velocity profile.

### 4.3 Overview of MRI Gradients

#### 4.3.1 Control Module

It is important to know the trajectory, the position of the MRbot, and gradients. Three set points need to be generated. A setpoint that contains the coordinate of points on the path centerline needs to be generated [46]. The velocity of the setpoint for the sphere is taken as the reference value. The trajectory controller is composed of a PID regulator and a feedforward component that directly outputs the optimal control. Errors on the velocity can be calculated with:

$$Error_v = V_c - V_s + K_r * (P_c - P_s) \quad (7)$$

where  $V_c$  is current velocity,  $V_s$  is velocity setpoint, and  $K_r = 0.7$  which defines how much error in position is corrected compared to the error in the velocity. The last two variables are  $P_c$ , which is the current position, and  $P_s$ , which is the position setpoint.

The PID regulator:

$$PF = -k_p * Error_v \quad (8)$$

$$PI = PI - (Error_v * \delta * k_i) \quad (9)$$

$$PD = -k_d * Error_{dt} \quad (10)$$

$$Error_{dt} = (Error_v - Error_p) / \delta \quad (11)$$

The parameters used for the PID controller are  $k_p = 2$ ,  $k_i = 1$ ,  $k_d = 0.01$ ,  $\delta = \text{base velocity}$ , and

$Error_p$  is errors on the previous position.

$$Moment_s = M * Vol \quad (12)$$

$$Vol = \frac{4}{3} \times \pi \times r^3 \quad (13)$$

where  $Moment_s$  is the moment sphere,  $M$  ( $1.9496 \times 10^6$  A/m) is the magnetization sphere,  $Vol$  is the volume sphere, and  $r$  (0.3 mm) is the sphere radius.

$FF$  is the optimal control that corresponds to the gradients of compensation for the drag force. It is produced by the blood:

$$FF = \frac{1}{2} \cdot C_d \cdot p \cdot R_e \cdot V \quad (14)$$

where  $C_d = 0.47$  is the drag coefficient,  $p$  is blood density,  $R_e$  is the frontal area of the sphere, and  $V$  is the blood velocity.

#### 4.3.2 Gradients Control

Gradients ( $G$ ) can be calculated with:

$$G = (1/Moment_s) * (PF + PI + PD + FF) \quad (15)$$

where  $Moment_s$  is the moment sphere,  $PF$ ,  $PI$ ,  $PD$  are PID controllers, and  $FF$  is the optimal control. See Figures 18, 19, 20. From the figures, we can see that there are some spikes in the gradients. There is one reason that may result in spikes. Because PID is a feedback system,  $PI$  always needs an initialization value to calculate the first PI, even if we avoid the case, which leads to the appearance of spikes.



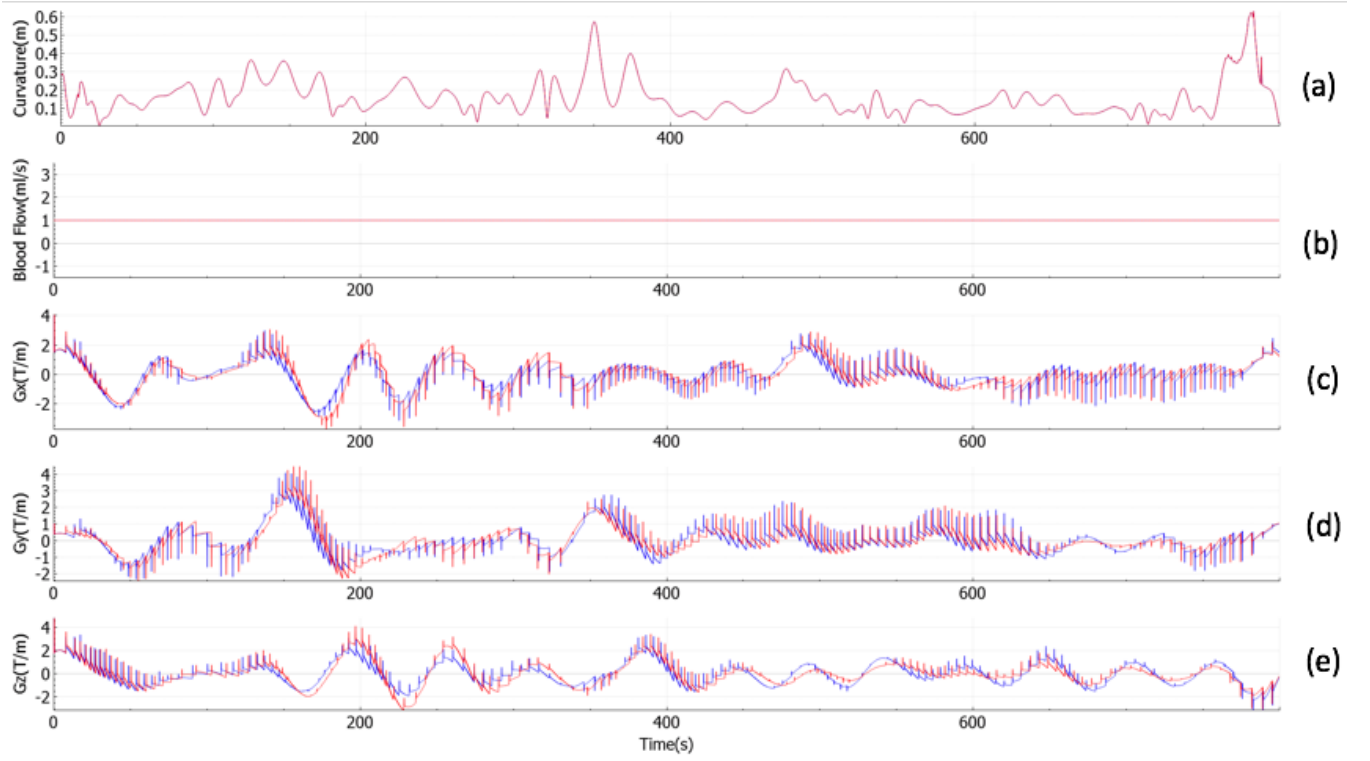


Figure 18: Results of  $T_p = 100$  ms (blue) and  $T_p = 200$  ms (red) of the simulation in the constant blood flow. (a) is a plot of the curvature of the path. (b) shows the constant blood flow. (c), (d), and (e) present the gradients generated by the MRI scanner for each axis.

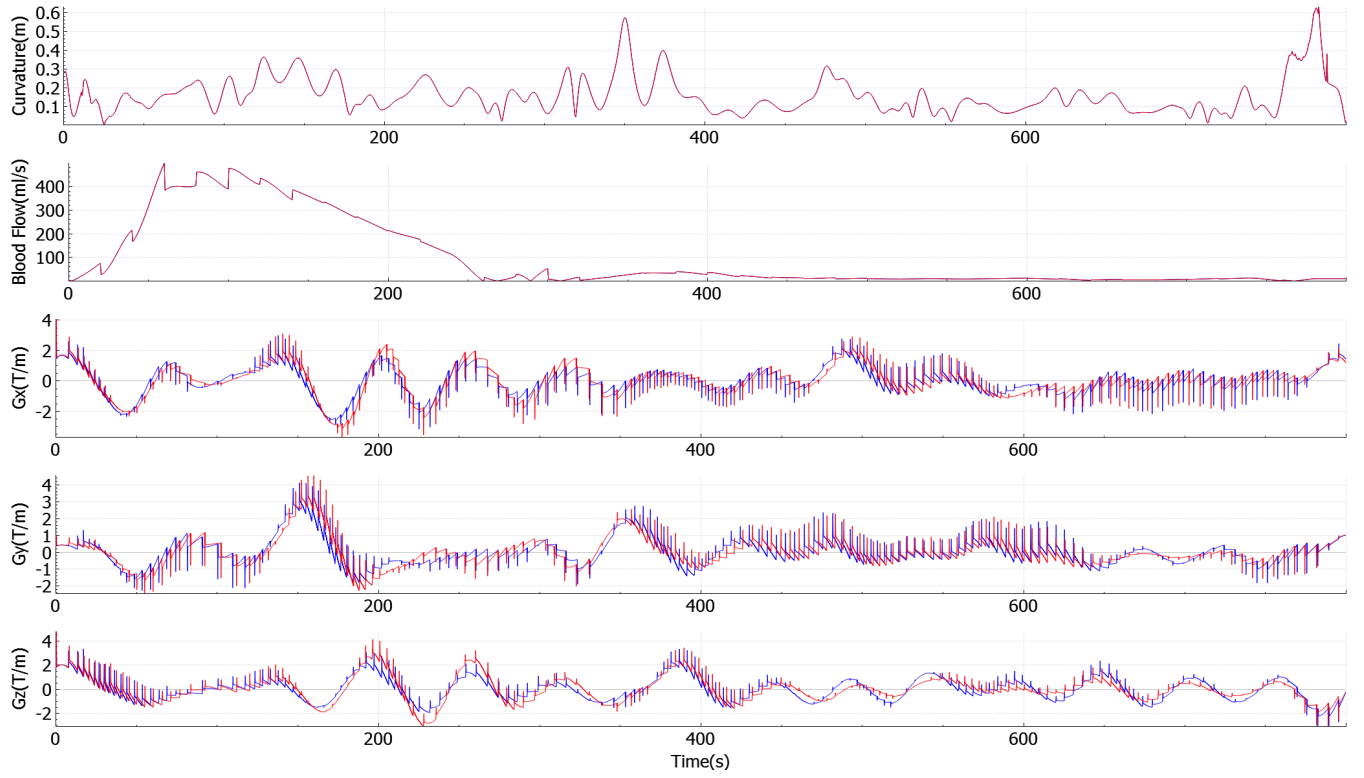


Figure 19: Results of  $T_p = 100$  ms (blue) and  $T_p = 200$  ms (red) of the simulation in the normal blood flow. (a) is a plot of the curvature of the path. (b) shows the normal blood flow. (c), (d), and (e) present the gradients generated by the MRI scanner for each axis.

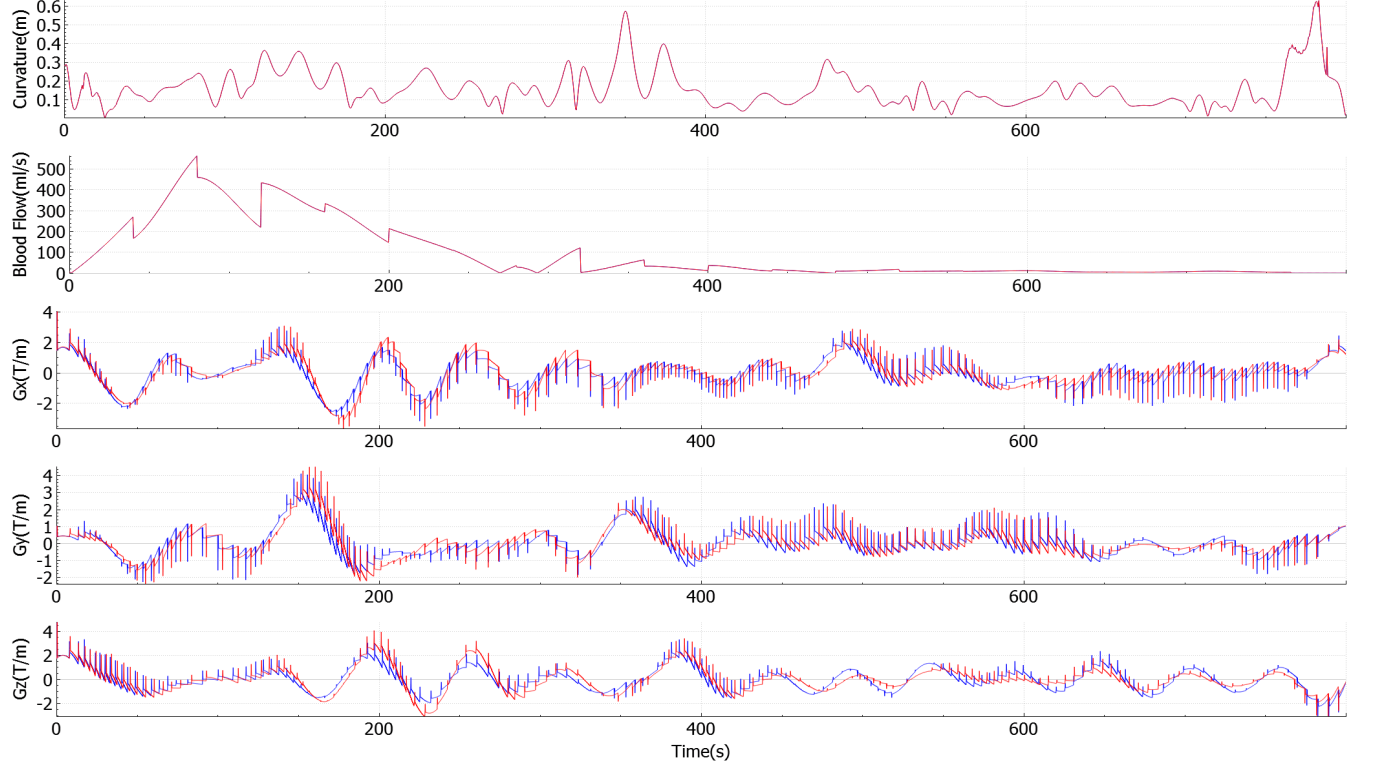


Figure 20: Results of  $T_p = 100$  ms (blue) and  $T_p = 200$  ms (red) of the simulation in fast heart rate of blood flow. (a) is a plot of the curvature of the path. (b) shows the fast heart rate blood flow. (c), (d), and (e) present the gradients generated by the MRI scanner for each axis.

#### 4.3.3 Magnetic Force

In order to get a strong magnetic force, the parameters to be maximized are the magnetization of the MRbot. Knowing that a ferromagnetic propulsion core is easy to fabricate and miniaturize as the viscosity of the blood is lower than cardiac tissues, the force is directly proportional to the ferromagnetic volume. Although the diameter of the blood vessels may vary due to cycles of heart contraction and the blood vessels, which affect drag force, our model considered the blood vessels as rigid cylindrical tubes.

$$F = M \cdot G \cdot V_o \quad (16)$$

Magnetic Resonance Propulsion (MRP) consists of applying magnetic gradients to exert a displacement force on MRbot. In (16),  $F$  is the magnetic force produced by magnetic gradients ( $T\ m^{-1}$ ),  $M$  is the magnetization of the material ( $A/m$ ),  $V_o$  is the volume of the ferromagnetic body ( $m^3$ ),  $G$  is the gradient or spatial variation of the magnetic induction. It can be seen from the equation that as the volume or the overall dimension of a ferromagnetic core in an MRbots or the volume of the ferromagnetic MRbots itself decreases, the force induced by the MRI system will decrease as well [47][48]. In order to produce sufficient magnetic force, using a ferromagnetic material with a stronger saturation magnetization is required. Friction between the sphere and tube walls is not considered in this study as gradients which are provided by MRI systems are strong enough to levitate MRbots.

#### 4.4 Simulation and Results

The built-in magnetic field strengths currently give tens of mT/m of gradients in any direction for imaging purposes. Such gradients produce a strong magnetic force in many directions. In order to correctly propel the ferromagnetic core on the pre-planned path and to provide a feel for this experiment, we measure the influence of magnetic field gradients from an MRI system on a ferromagnetic sphere. This ferromagnetic field is suitable because permendur has strong magnetization with a saturation value that is higher than other magnetic materials. In the computational core, each position has three gradients  $[X(i), Y(i), Z(i)]$  ( $i$  refers to the digitized form) on the 3D vascular path. The navigation of the ferromagnetic sphere relies mainly on two aspects: drag force and the propulsion magnetic force. During the experiments, robust software architecture must be present to seamlessly and precisely compute and apply the necessary gradients information. We ran the experiment 1,000 times and record the results. The execute time is within 7.2 ms to 8.1 ms and the average running time is 7.7 ms. We also calculated the spatial gradient to obtain more information. For example, in the X-axis: 0.93 mT/m, 1.68 mT/m, 3.35 mT/m, and 3.038 mT/m, in the Y-axis: 3.33 mT/m, 1.13 mT/m, 3.84 mT/m, and 3.53 mT/m, in the Z-axis: 2.65437 mT/m, 3.7658 mT/m, 3.61 mT/m, and 2.84 mT/m. Such gradients can be used to produce a magnetic force vector in

upward and downward directions. We assume blood density is constant which means that we can neither increase nor decrease the amount of blood in the vessel. The volume of blood that goes in must equal the volume that comes out.

#### 4.4.1 Time-varying Magnetic Fields

The gradient coils are used for imaging produce time-varying magnetic fields slew rate in dB/dt [49]. During the rise time of the magnetic field, an electric current may be induced in a conductor. The human body is a conductor which can produce peripheral nerve stimulation. This stimulation may result in a slight tingling sensation or a brief muscle twitch but is not recognized as a significant health risk [50]. In order to avoid these potential hazards, the FDA has the rule to guide gradient field switch if dB/dt is sufficient to produce propulsion in vessels. The slew-rate dB/dt is limited by the FDA and is generally below 200 T/m/s. Slew rates can be calculated with [49]:

$$S = G_p / T_r * r \quad (17)$$

where  $S$  is slew rates (dB/dt),  $G_p$  is the difference between two consecutive gradients,  $T_r$  is rise time, and  $r$  is the distance of 50 cm from the isocenter during the scan. dB/dt is measured in units of Tesla per meter per second (T/m/s). The slew rates and gradient strengths above are  $x$ -gradient,  $y$ -gradient, and  $z$ -gradient. Our results are safe, within the FDA threshold. See Figures 21, 22, 23.

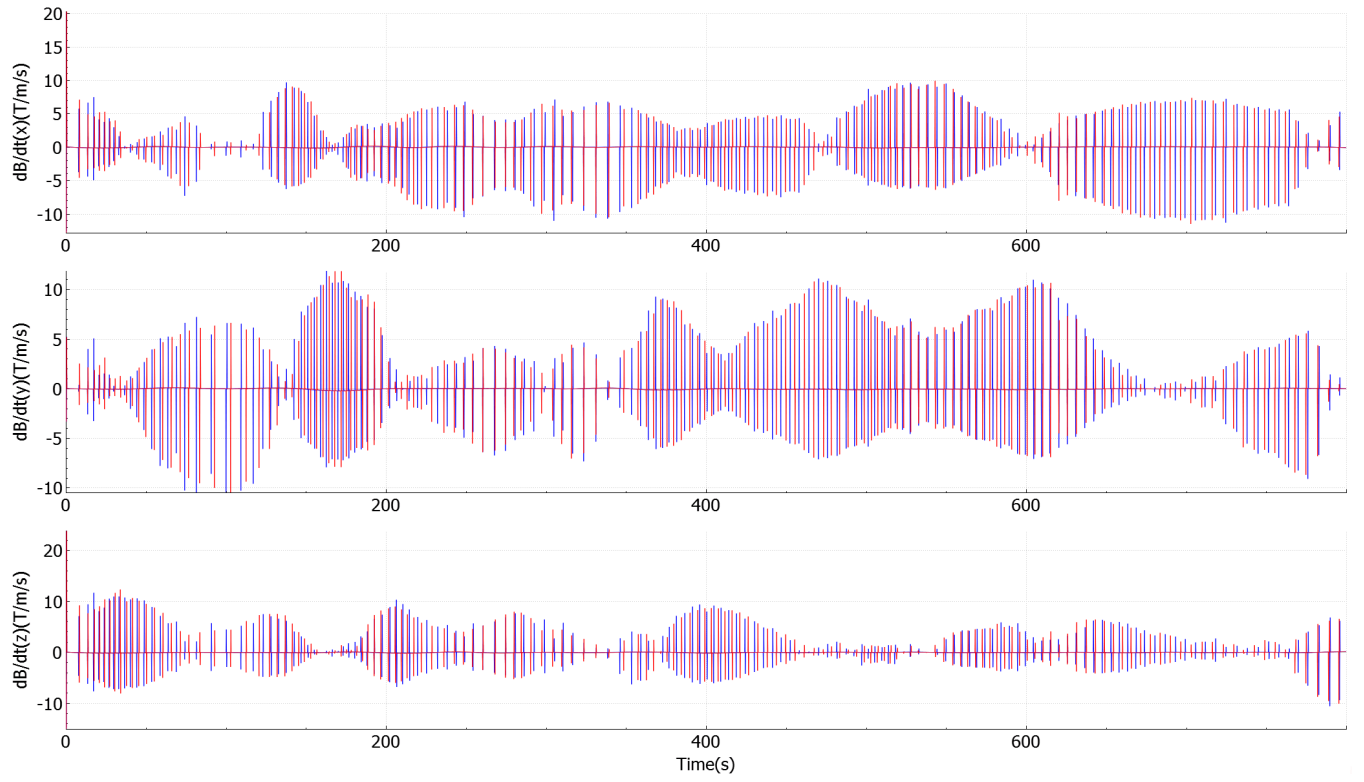


Figure 21: Results of  $T_p = 100$  ms (blue) and  $T_p = 200$  ms (red) of the simulation in the constant blood flow. (a), (b), and (c) show  $dB/dt$  for each axis. The blood flow is 1 ml/s.

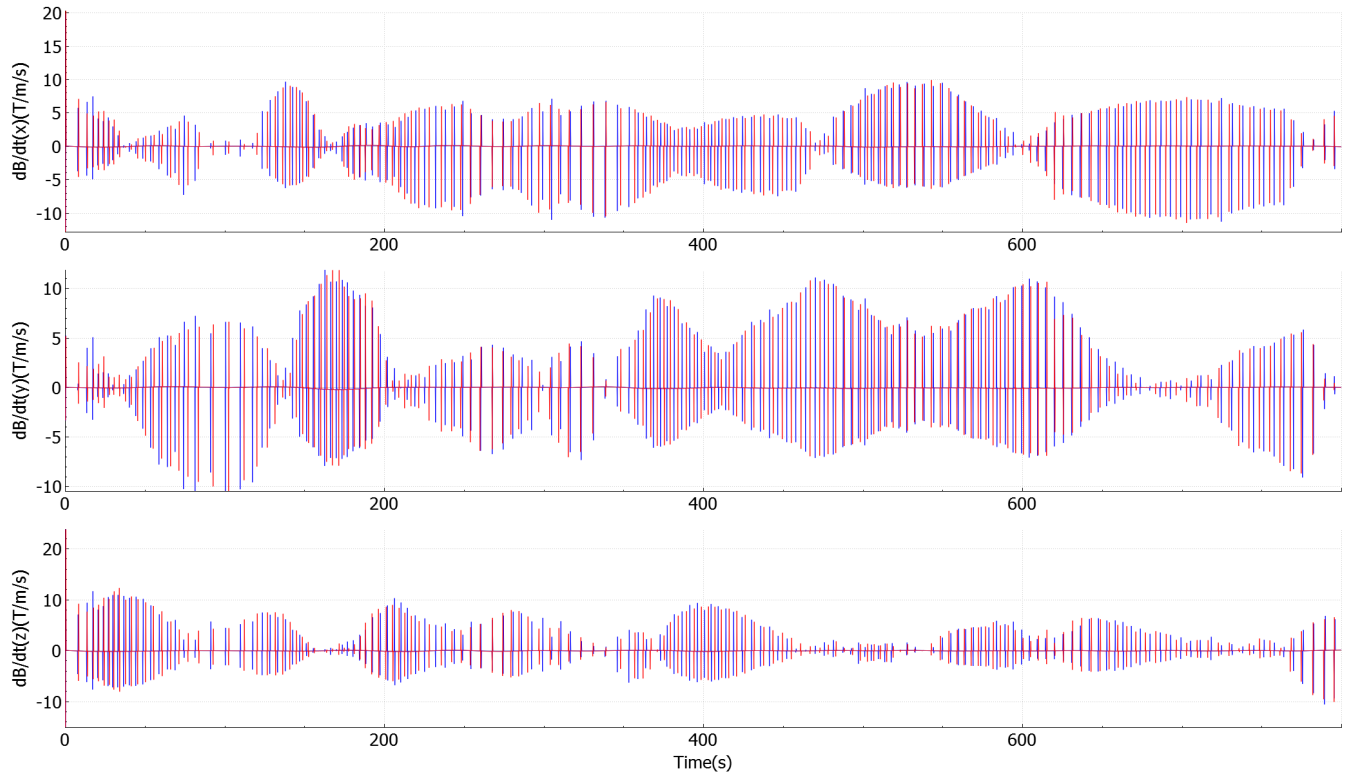


Figure 22: Results of  $T_p = 100$  ms (blue) and  $T_p = 200$  ms (red) of the simulation in the normal blood flow. (a), (b), and (c) show  $dB/dt$  for each axis.

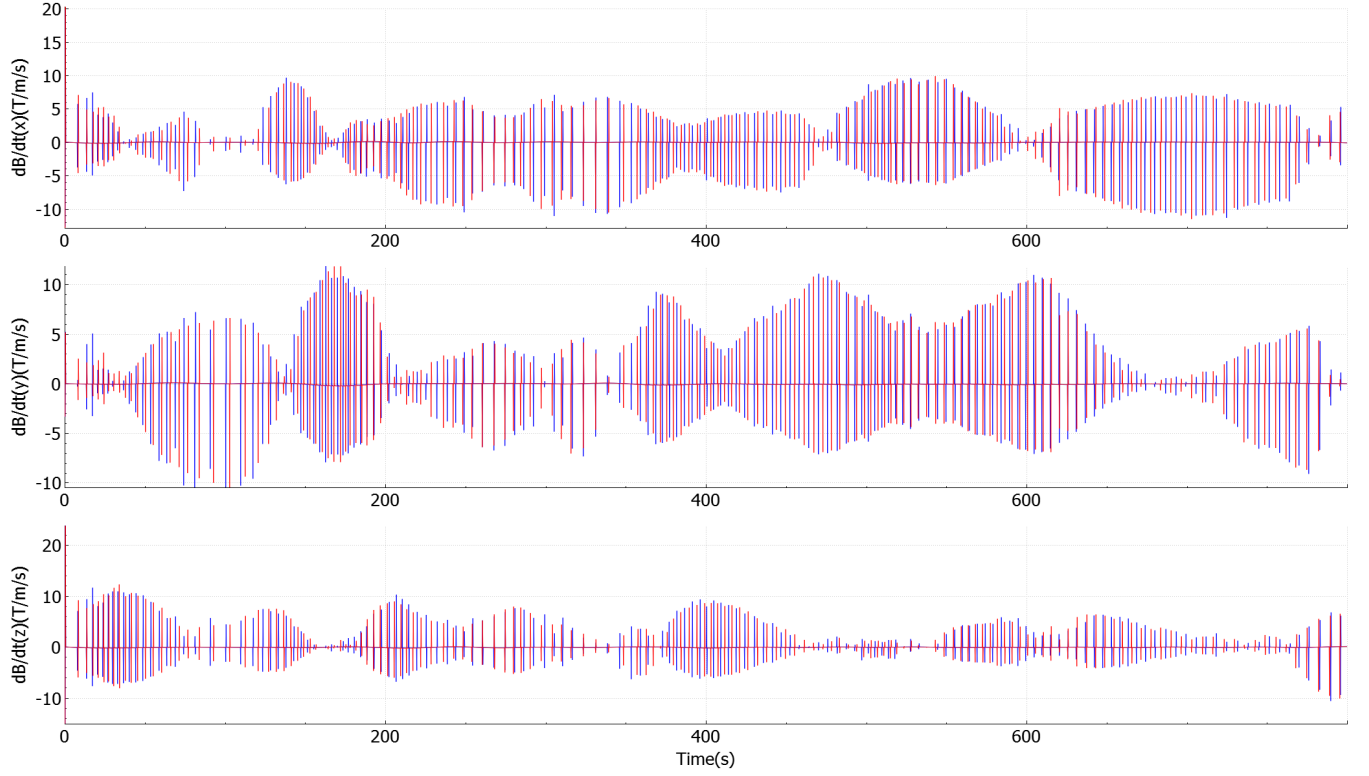


Figure 23: Results of  $T_p = 100$  ms (blue) and  $T_p = 200$  ms (red) of the simulation in the fast heart rate of blood flow. (a), (b), and (c) show  $dB/dt$  for each axis.

#### 4.4.2 Visualization Toolkit

Our approach is to augment an existing visualization system. The VTK is a freely available C++ class library for 3D graphics and visualization [51]-[53]. When we get each coordinate of points, we want to create a sharply focused object library that could easily embed in our sphere pathway. The sphere pathway can be built from small pieces. The key to toolkits is that pieces must be well defined, with simple interfaces.

#### 4.4.3 Simulation

The ability to handle a large dataset is a critical requirement. From past studies, we see that ParaView is an open-source application with multi-platform visualization and scales well [54]. This



tool provides a graphical user interface for the interactive exploration of large data sets. Our framework is built on top of ParaView, an analysis optimized C++ designed and modern visualization tool in post-processing, for advanced modeling and simulation codes.

Qt is one of the most important cross-platform application frameworks that is used for the development of GUI programs. We developed web-enabled applications for deploying them across personal computer desktops, mobile devices, and embedded operating systems without rewriting the source code. See Figure 24. Red and white lines represent actual sphere maps and desired paths. The red path has 8380 experimental data points when the sphere moved along a selected path. Currently, the step size taken is 0.0001 and this small step size can be increased and determined to a value where we do not lose much accuracy. The MRbot closely follows the vascular trajectory.

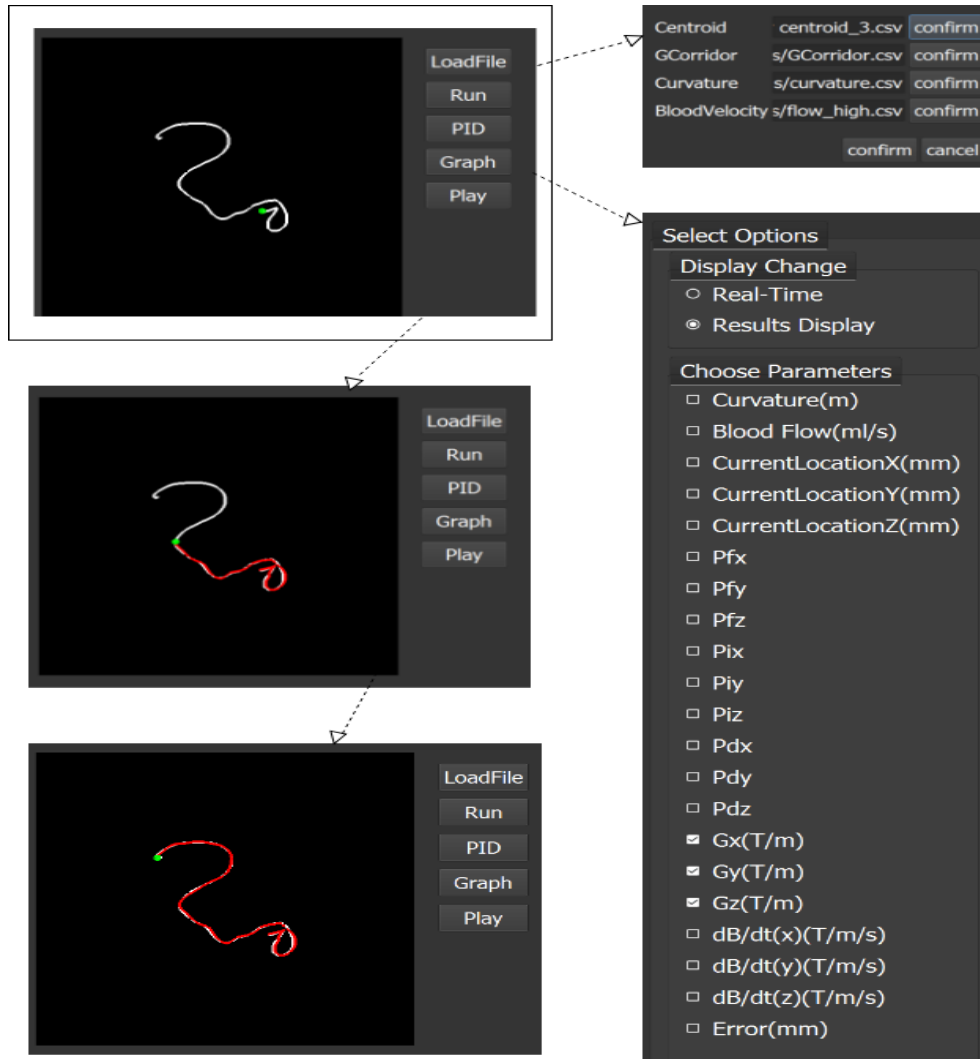


Figure 24: Structure of Qt user interface

## 5 Conclusion

In this research, we provided a comprehensive overview of three deep learning approaches for detecting and segmenting left ventricles with a fast training speed. Firstly, we presented the BNU-net architecture for constructing, training, and testing with batch-normalized. Experimental results show that our methodology produces the best results in each one of the measures used in the database of the MICCAI grand challenge (dice metric, sensitivity, and perpendicular distance) compared with other groups' results. Secondly, we proposed two deep learning architectures called LNU-net and IBU-net for left ventricle segmentation. By combining multiple models trained with the Sunnybrook dataset, our proposed models performed better performance of segmentation accuracy. In addition, we identified the IBU-net network with instance-batch normalization that the inputs stabled and shifted to maintain the network. This further verifies a real-time implementation of the proposed method is sufficiently fast to be used for intraoperative registration of preoperative cardiac anatomy [55].

The major contributions by modifying U-net in this study are listed as follows. First of all, the normalization method was effective for improving dice scores in image segmentation tasks. Among the widely-used normalization methods, we found that batch instance normalization was better to perform segmentation. Secondly, ELU showed a better performance than ReLU, which led to a slight but convincing improvement. With the combination of batch instance normalization and ELU, we got a state-of-the-art segmentation dice mean of 0.96. Last but not least, the training speed was accelerated with our current methods. The training time per epoch was reduced from 12s to 9s. Compared to the original version of U-net, our version displayed a higher dice score and faster speed, which means we got better results with fewer computational resources. We hope our work could be helpful for the practical image segmentation work in medical applications. We intend for this work to relieve both experienced doctors and recovering patients in their fight against cardiac diseases.

In this research, we also simulated a ferromagnetic core inside a vessel by exploiting the real-time

features. The core is implemented in C/C++ for streaming development, testing, and processing the output. In the work, appropriate libraries (e.g., VTK) and two realistic blood flow waveforms are incorporated in this simulation. We manipulated the MRbot with predefined paths which achieve high accuracy propulsion and transport. Because of the experimental studies that require a real-time operation, the platform was implemented on the Qt framework with software modules performing specific tasks running on dedicated threads: PID controller, generation of VF, and generation of MR gradient waveforms.

Deep learning-based methods for the fully automated left ventricle segmentation from short-axis cine MR images can achieve good accuracy. However, with low quality of images, these methods can still fail on medical image segmentation. Future studies can be included: improving the quality of images, developing image processing algorithms, and developing algorithms that can give instant feedback to correct and optimize the image acquisition process.

## Bibliography

- [1] R. P. K. Poudel, P. Lamata, and G. Montana. Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation. *Lecture Notes in Computer Science*, pp. 83–94, 2017.
- [2] P. V. Tran. A fully convolutional neural network for cardiac segmentation in short-axis MRI. arXiv preprint arXiv: 1604.00494, 2016.
- [3] T. A. Ngo and G. Carneiro. Left ventricle segmentation from cardiac MRI combining level set methods with deep belief networks. *2013 IEEE International Conference on Image Processing*, pp. 695-699, 2013.
- [4] S. Queirós, D. Barbosa, B. Heyde, P. Morais, J. L. Vilça, D. Friboulet, O. Bernard, and J. D’hooge. Fast automatic myocardial segmentation in 4D cine CMR datasets. *Medical Image Analysis*, vol. 18, issue. 7, 2014.
- [5] H. Hu, H. Liu, Z. Gao, and L. Huang. Hybrid segmentation of left ventricle in cardiac MRI using Gaussian-mixture model and region restricted dynamic programming. *Magnetic Resonance Imaging*, 31(4), pp. 575-584, Dec. 2012.
- [6] S. Huang, J. Liu, L. C. Lee, S. K. Venkatesh, L. L. S. Teo, C. Au, and W. L. Nowinski. An image-based comprehensive approach for automatic segmentation of left ventricle from cardiac short axis cine MR images. *Journal of Digital Imaging*, pp. 598-608, 2011.
- [7] J. Margeta, E. Geremia, A. Criminisi, N. Ayache, and A. Criminisi. Layered spatio-temporal forests for left ventricle segmentation from 4D cardiac MRI data. *MICCAI workshop: Statistical Atlases and Computational Models of the Heart (STACOM)*, Sep. 2012.
- [8] H. Liu, H. Hu, X. Xu, and E. Song. Automatic left ventricle segmentation in cardiac MRI using topological stable-state thresholding and region restricted dynamic programming. *Academic Radiology*, 19(6):723-731, 2012.
- [9] N. V. Navkar, Z. Deng, D. J. Shah, K. E. Bekris and N. V. Tsekos. Visual and force-feedback guidance for robot-assisted interventions in the beating heart with real-time MRI. *2012 IEEE*

International Conference on Robotics and Automation, pp. 689-694, 2012.

[10] M. Vonthron, V. Lalande, G. Bringout, C. Tremblay and S. Martel. A MRI-based integrated platform for the navigation of micro-devices and microrobots. 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1285-1290, Sep. 2011.

[11] E. Yeniaras, N. V. Navkar, A. E. Sonmez, D. J. Shah, Z. Deng, and N. V. Tsekos. MR-based real time path planning for cardiac operations with transapical access. In International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), 14(Pt 1): 25-32, 2011.

[12] S. Martel. Magnetic navigation control of microagents in the vascular network: challenges and strategies for endovascular magnetic navigation control of microscale drug delivery carriers. IEEE Control Systems Magazine, vol. 33, no. 6, pp. 119-134, Dec. 2013.

[13] M. Latulippe and S. Martel. A progressive multidimensional particle swarm optimizer for magnetic core placement in dipole field navigation. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2314-2320, Oct. 2016.

[14] A. H. E. M. Maas and Y. E. A. Appelman. Gender differences in coronary heart disease. Netherlands Heart Journal : Monthly Journal of the Netherlands Society of Cardiology and the Netherlands Heart Foundation, 18(12), pp. 598-602, 2010.

[15] A. A. Inamdar and A. C. Inamdar. Heart failure: diagnosis, management and utilization. Journal of Clinical Medicine, 5(7): 62, Jun. 2016.

[16] P. L. Kerkhof. Characterizing heart failure in the ventricular volume domain. Clinical Medicine Insights: Cardiology, vol. 9, (Suppl 1): 11-31, Feb. 2015.

[17] W. Wang, Y. Wang, Y. Wu, T. Lin, S. Li and B. Chen. Quantification of full left ventricular metrics via deep regression learning with contour-guidance. In IEEE Access, vol. 7, pp. 47918-47928, 2019.

[18] J. Cong, Y. Zheng, W. Xue, B. Cao and S. Li. Ma-shape: modality adaptation shape regression

- for left ventricle segmentation on mixed MR and CT images. In *IEEE Access*, vol. 7, pp. 16584-16593, 2019.
- [19] C. Shorten and T. M. Khoshgoftaar. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, vol 6, no. 1, 2019.
- [20] C. Petitjean, J. Dacher. A review of segmentation methods in short axis cardiac MR images. *Medical Image Analysis*, vol. 15, pp. 169–184, 2011.
- [21] O. Ronneberger, P. Fischer, and T Brox. U-net: convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234-241, 2015.
- [22] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9, Boston, MA, USA, 2015.
- [23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations*, San Diego, CA, USA, May. 2015.
- [24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, vol. 1, pp. 1097-1105, Curran Associates Inc., Red Hook, NY, USA, 2012.
- [25] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, Las Vegas, NV, USA, 2016.
- [26] S. Ioffe and C. Szegedy. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, vol. 37, pp. 448-456, 2015.
- [27] S. Liu, Y. Wang, X. Yang, B. Lei, L. Liu, S. X. Li, D. Ni, and T. Wang. Deep learning in medical ultrasound analysis: a review. *Engineering*, vol. 5, issue 2, pp. 261-275, 2019.

- [28] X. Zhou and G. Yang. Normalization in training U-net for 2-D biomedical semantic segmentation. *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1792-1799, 2019.
- [29] I. Croitoru, S. V. Bogolin, and M. Leordeanu. Unsupervised learning of foreground object segmentation. *International Journal of Computer Vision*, vol 127, no 9, pp. 1279-1302, 2019.
- [30] P. Y. Simard, D. Steinkraus, J. C. Platt. Best practices for convolutional neural networks applied to visual document analysis. *7th International Conference on Document Analysis and Recognition, Proceedings*, pp. 958-963, Edinburgh, UK, 2003.
- [31] W. Chu, G. Molina, N. V. Navkar, C. F. Eick, A. T. Becker, P. Tsiamyrtzis, and N. V. Tsekos. BNU-net: a novel deep learning approach for lv MRI analysis in short-axis MRI. In *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 731-736, Athens, Greece, 2019.
- [32] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv: 1607.06450*, Jul.2016.
- [33] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: the missing ingredient for fast stylization. *arXiv preprint arXiv: 1607.08022*, Jul. 2016.
- [34] H. Nam and H. E. Kim. Batch-instance normalization for adaptively style-invariant neural networks. In *Advances in Neural Information Processing Systems*, pp. 2558-2567, 2018.
- [35] Sunnybrook Cardiac Data. <https://www.cardiacatlas.org/studies/sunnybrook-cardiac-data/>
- [36] X. Y. Zhou, P. Li, Z. Y. Wang, and G. Z. Yang. U-net training with instance-layer normalization. In *International Workshop on Multiscale Multimodal Medical Imaging*, pp. 101-108, Springer, 2019.
- [37] J. B. Mathieu, S. Martel, L. Yahia, G. Soulez, and G. Beaudoin. MRI systems as a mean of propulsion for a microdevice in blood vessels. *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 4, pp. 3419-3422, Sep. 2003.



- [38] S. Martel, J. B. Sylvain, O. Felfoul, H. Macicior, G. Beaudoin, G. Soulez, L. H. Yahia. Adapting MRI systems to propel and guide microdevices in the human blood circulatory system. Annual International Conference of the IEEE Engineering in Medicine and Biology Society, vol 2, pp. 2044-2047, 2004.
- [39] T. Thom, N. Haase, W. Rosamond, V. J. Howard, J. Rumsfeld, T. Manolio, Z. J. Zheng, K. Flegal, C. O'Donnell, S. Kittner, D. Lloyd-Jones, D. C. Goff, Y. Hong, R. Adams, G. Friday, K. Furie, P. Gorelick, B. Kissela, J. Marler, J. Meigs, V. Roger, S. Sidney, P. Sorlie, J. Steinberger, S. W. Smoller, M. Wilson, and P. Wolf. Heart disease and stroke statistics–2006 update: a report from the American Heart Association Statistics Committee and Stroke Statistics Subcommittee. The American Heart Association Statistics Committee and Stroke Statistics Subcommittee, 2006.
- [40] Y. Shi, P. Lawford, and R. Hose. Review of Zero-D and 1-D models of blood flow in the cardiovascular system. BioMedical Engineering OnLine, vol. 10, Apr. 2011.
- [41] F. M. White. Viscous fluid flow. ser Mechanical Engineering, McGraw-Hill, New York, NY, USA, 1991.
- [42] R. Kehlenbeck and R. D. Felice. Empirical relationships for the terminal settling velocity of sheres in cylindrical columns. Chemical Engineering & Technology, vol. 22, pp. 303-308, 1999.
- [43] M. Sitti, H. Ceylan, W. Hu, J. Giltinan, M. Turan, S. Yim and E. Diller. Biomedical applications of untethered mobile milli/microrobots. Proceedings of the IEEE, vol 103, pp. 205-224, 2015.
- [44] M. Sendoh, K. Ishiyama, K. I. Arai, M. Jojo, F. Sato and H. Matsuki. Fabrication of magnetic micro-machine for local hyperthermia. 2002 IEEE International Digest of Technical Papers on Magnetism Conference, pp. fu11, Apr. 2002.
- [45] R. G. McNeil, R. C. Ritter, B. Wang, M. A. Lawson, G. T. Gillies, K. G. Wika, E. G. Quale, M. A. Howard 3rd, and M. S. Grady. Functional design features and initial performance characteristics of a magnetic-implant guilience system for stereotactic neurosurgery. In IEEE Transactions on Biomedical Engineering, vol. 42, pp. 793-801, Aug. 1995.

- [46] M. S. Grady, M. A. Howard 3rd, J. A. Molloy, R. C. Ritter, E. G. Quate, and G. T. Gillies. Preliminary experimental investigation of in vivo magnetic manipulation: results and potential application in hyperthermia. *Medical Physics*, vol. 16, pp. 263-272, Mar. 1989.
- [47] J. A. Molloy, R. C. Ritter, M. S. Grady, M. A. Howard 3rd, E. G. Quate, and G. T. Gillies. Experimental determination of the force required for insertion of a thermoseed into deep cardiac tissues. *Annals of Biomedical Engineering*, vol. 18, pp. 299-313, 1990.
- [48] P. H. T. Uhlherr and R. P. Chhabra. Wall effect for the fall of spheres in cylindrical tubes at high reynolds number. *Canadian Journal of Chemical Engineering*, vol. 73, pp. 918-923, 1995.
- [49] J. B. Mathieu, S. Martel, L. Yahia, G. Soulez, and G. Beaudoin. Preliminary investigation of the feasibility of magnetic propulsion for future microdevices in blood vessels. *Bio-medical Materials and Engineering*, vol. 15, pp. 367-374, Feb. 2005.
- [50] Y. Saadeh and D. Vyas. Nanorobotic applications in medicine: current proposals and designs. *American Journal of Robotic Surgery*, 1(1): 4-11, 2014.
- [51] W. Schroeder, K. M. Martin and W. E. Lorensen. The visualization toolkit: an object-oriented approach to 3D graphics. Prentice-Hall, Inc., Hoboken, NJ, USA, 2 edition, 1998.
- [52] J. Ahrens, S. Jourdain, P. OLeary, J. Patchett, D. H. Rogers and M. Petersen. An image-based approach to extreme scale in situ visualization and analysis. *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pp. 424-434, New Orleans, LA, USA, 2014.
- [53] W. J. Schroeder, K. M. Martin, and W. E. Lorensen. The design and implementation of an object-oriented toolkit for 3D graphics and visualization. IEEE Computer Society Press, pp. 93-ff., San Francisco, CA, USA, 1996.
- [54] J. P. Ahrens, B. Geveci, and C. C. Law. ParaView: an end-user tool for large data visualization. *The Visualization Handbook*, 2005.
- [55] X. Gao, N. V. Navkar, D. J. Shah, N. V. Tsekos and Z. Deng. Intraoperative registration

- of preoperative 4D cardiac anatomy with real-time MR images. 2012 IEEE 12th International Conference on Bioinformatics Bioengineering (BIBE), pp. 583-588, Larnaca, Cyprus, 2012.
- [56] T. Kensicher, J. Leclerc, D. Biediger, D. J. Shah, I. Seimenis, A. T. Becker, N. V. Tsekos. Towards MRI-guided and actuated tetherless milli-robots: preoperative planning and modeling of control. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 6440-6447, 2017.
- [57] J. P. V. Batavia and P. Casale. Robotic surgery in pediatric urology. *Current Urology Reports*, 15(5), 402, 2014.
- [58] F. Cepolina, B. Challacombe, and R. C. Michelini. Trends in robotic surgery. *Journal of Endourology*, 19(8): pp. 940-951, 2005.
- [59] W. R. Chitwood, A. P. Kypson, and L. Nifong. Robotic mitral valve surgery: a technologic and economic revolution for heart centers. *American Heart Hospital Journal*, 1(1): pp. 30-39, 2003.
- [60] O. Felfoul, J.-B. Mathieu, G. Beaudoin, and S. Martel. In vivo MR-tracking based on magnetic signature selective excitation. *IEEE Transactions on Medical Imaging*, 27(1): pp. 28-35, 2008.
- [61] S. Martel, J. Mathieu, O. Felfoul, A. Chanu, E. Aboussouan, S. Tamaz, P. Pouponneau, L. Yahia, G. Beaudoin, G. Soulez, and M. Mankiewicz. Automatic navigation of an untethered device in the artery of a living animal using a conventional clinical magnetic resonance imaging system. *Applied Physics Letters*, 90(11): 114105, 2007.
- [62] T. Liu and M. C. Cavaşoğlu. Three dimensional modeling of an MRI actuated steerable catheter system. 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 4393-4398, 2014.
- [63] Y. Zhao, W. Gui, C. Cheng, J. Tang, and L. Li. Medical images edge detection based on mathematical morphology. 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, pp. 6492-6495, 2005.
- [64] C. Cigla and A. A. Alatan. Region-based image segmentation via graph cuts. In *Proceedings*

of the 2008 15th IEEE International Conference on Image Processing, pp. 2272–2275, San Diego, CA, USA, Oct. 2008.

[65] A. Xu, L. Wang, S. Feng, and Y. Qu. Threshold-based level set method of image segmentation. 2010 Third International Conference on Intelligent Networks and Intelligent Systems, pp. 703–706, 2010.

[66] F. Milletari, N. Navab and S. Ahmadi. V-Net: fully convolutional neural networks for volumetric medical image segmentation. Fourth International Conference on 3D Vision (3DV), pp. 565–571, 2016.

[67] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3D U-net: learning dense volumetric segmentation from sparse annotation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Athens, Greece, pp. 424–432, Oct. 2016.

[68] Z. Wang, N. Zou, D. Shen, and S. Ji. Non-local U-nets for biomedical image segmentation. In Proceedings of the AAAI, pp. 6315–6322, New York, NY, USA, Feb. 2020.

[69] O. Oktay, J. Schlemper, L. L. Folgoc, M. J. Lee, M. Heinrich, K. Misawa, K. Mori, S. G. McDonagh, N. Hammerla, B. Kainz, B. Glocker, and D. Rueckert. Attention U-net: learning where to look for the pancreas, arXiv preprint arXiv: 1804.03999, 2018.

[70] N. Ibtehaz and M. S. Rahman. MultiResUNet: rethinking the U-net architecture for multi-modal biomedical image segmentation. Neural Networks, pp. 74–87, 2020.

[71] M. Lynch, O. Ghita, and P. F. Whelan. Automatic segmentation of the left ventricle cavity and myocardium in MRI data. Computers in Biology and Medicine, vol. 36, pp. 389–407, 2006.

[72] C. Cassen, J. Domenger, J. Braquelaire, and J. L. Barat. Left ventricular segmentation in MRI images. Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis, pp. 244–249, 2001.

[73] A. Katouzian, A. Prakash, E. Konofagou. A new automated technique for left-and right-ventricular segmentation in magnetic resonance imaging. Annual International Conference of the

IEEE Engineering in Medicine and Biology Society, pp. 3074–3077, 2006.

[74] J. Lötjönen, S. Kivistö, J. Koikkalainen, D. Smutek, and K. Lauerma. Statistical shape model of atria, ventricles and epicardium from short-and long-axis MR images. *Medical Image Analysis*, vol. 8, no. 3, pp. 371–386, 2004.

[75] A. Pednekar, U. Kurkure, R. Muthupillai, S. Flamm, and I. A. Kakadiaris. Automated left ventricular segmentation in cardiac MRI. *IEEE Transactions on Bio-medical Engineering*, vol. 53, no. 7, pp. 1425–1428, 2006.

[76] S. Charmchi, K. Punithakumar and P. Boulanger. Optimizing U-net to segment left ventricle from magnetic resonance imaging. 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 327–332, 2018.

[77] G. E. Hinton. Learning multiple layers of representation. *Trends in Cognitive Sciences*, vol. 11, no. 10, pp. 428–434, 2007.

[78] D. C. Cireşan, U. Meier, L. M. Gambardella and J. Schmidhuber. Deep, big, simple neural nets for handwritten digit recognition. In *Neural Computation*, vol. 22, pp. 3207–3220, 2010.

[79] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Conference on Computer Vision and Pattern Recognition*. pp. 3431–3440, 2015.

[80] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang. Unet++: a nested U-net architecture for medical image segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 3–11, Springer, 2018.

[81] L. Spirkovska. A summary of image segmentation techniques. 1993.

[82] V. Dey, Y. Zhang, and M. Zhong. A review on image segmentation techniques with remote sensing perspective. *ISPRS TC VII Symposium - 100 Years ISPRS*, vol. 38, Vienna, Austria, 2010.

[83] D. L. Pham, C. Xu, and J. L. Prince. Current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, vol. 2, no. 1, pp. 315–337, 2000.

- [84] S. Lankton and A. Tannenbaum. Localizing region-based active contours. *IEEE transactions on Image Processing: A Publication of the IEEE Signal Processing Society*, vol. 17, no. 11, pp. 2029–2039, Nov. 2008.
- [85] K. S. Tan and N. A. M. Isa. Color image segmentation using histogram thresholding–fuzzy c-means hybrid approach. *Pattern Recognition*, vol. 44, pp. 1-15, Jan. 2011.
- [86] P. Dollar, Z. Tu, and S. Belongie. Supervised learning of edges and object boundaries. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1964-1971, Feb. 2006.
- [87] T. N. A. Nguyen, J. Cai, J. Zhang and J. Zheng. Robust interactive image segmentation using convex active contours. In *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3734-3743, Aug. 2012.
- [88] Z. Ju, J. Zhou, X. Wang, and Q. Shu. Image segmentation based on adaptive threshold edge detection and mean shift. *2013 IEEE 4th International Conference on Software Engineering and Service Science*, pp. 385-388, 2013.
- [89] T. Cour, F. Benezit, and J. Shi. Spectral segmentation with multiscale graph decomposition. In *Computer Vision and Pattern Recognition*, vol. 2, pp. 1124–1131, 2005.
- [90] L. Wang, H. Wu, and C. Pan. Region-based image segmentation with local signed difference energy. *Pattern Recognition Letters*, vol. 34, pp. 637-645, 2013.
- [91] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 603–619, 2002.
- [92] K. E. Melkemi, M. Batouche, and S. Fofou. A multiagent system approach for image segmentation using genetic algorithms and extremal optimization heuristics. *Pattern Recognition Letters*, vol. 27, no. 11, pp. 1230-1238, Aug. 2006.
- [93] P. Arbeláez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation

- using regions and parts. 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3378-3385, 2012.
- [94] Y. W. Chen and Y. Q. Chen. Adaptive image segmentation using artificial co-evolving tribes. *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 21, no. 7, pp. 1171, 2007.
- [95] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 898-916, 2011.
- [96] P. Huang, H. Cao, and S. Luo. An artificial ant colonies approach to medical image segmentation. *Computer Methods and Programs in Biomedicine*, vol. 92, no. 3, pp. 267–273, 2008.
- [97] Y. Han and P. Shi. An improved ant colony algorithm for fuzzy clustering in image segmentation. *Neurocomputing*, vol. 70, no. 4-6, pp. 665–671, 2007.
- [98] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. in *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.
- [99] P. Welandar, S. Karlsson, and A. Eklund. Generative adversarial networks for image-to-image translation on multi-contrast mr images-a comparison of cyclegan and unit. *arXiv preprint arXiv:1806.07777*, 2018.
- [100] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-cnn: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.
- [101] K. Andrej and F. Li. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3128–3137, 2015.
- [102] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

- [103] Y. LeCun. Generalization and network design strategies. *Connectionism in Perspective*, pp. 143–155, 1989.
- [104] Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time series. *The Handbook of Cardiac Theory and Neural Networks*, vol. 3361, no. 10, pp. 1995, 1995.
- [105] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.
- [106] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in Neural Information Processing Systems*, pp. 2843–2851, 2012.
- [107] A. D. Brebisson and G. Montana. Deep neural networks for anatomical cardiac segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 20–28, 2015.
- [108] Activation function : <https://medium.com/@shrutijadon10104776>.
- [109] 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*, vol. 15, pp. 1929–1958, 2014.
- [110] D. Gillick, C. Brunk, O. Vinyals, and A. Subramanya. Multilingual language processing from bytes. *arXiv preprint arXiv: 1512.00103*, 2015.
- [111] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu. Exploring the limits of language modeling. *arXiv preprint arXiv: 1602.02410*, 2016.
- [112] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. in *Advances in Neural Information Processing Systems*, pp. 3104–3112, 2014.
- [113] C. Chen, C. Qin, H. Qiu, G. Tarroni, J. Duan, W. Bai, and D. Rueckert. Deep learning for cardiac image segmentation: a review. *Frontiers in Cardiovascular Medicine*, vol. 7, 2020.



- [114] M. R. Avendi, A. Kheradvar, and H. Jafarkhani. A combined deep-learning and deformable-model approach to fully automatic segmentation of the left ventricle in cardiac MRI. *Medical Image Analysis*, vol. 30, pp. 108–119, 2016.
- [115] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, vol. 9, pp. 1735–1780, 1997.
- [116] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing (ACL)*, pp. 1724–1734, 2014.
- [117] X. Zhuang, L. Li, C. Payer, D. Stern, M. Urschler, M. P. Heinrich, J. Oster, C. Wang, Ö. Smedby, C. Bian, X. Yang, P-A. Heng, A. Mortazi, U. Bagci, G. Yang, C. Sun, G. Galisot, J-Y. Ramel, T. Brouard, Q. Tong, W. Si, X. Liao, G. Zeng, Z. Shi, G. Zheng, C. Wang, T. MacGillivray, D. Newby, K. Rhode, S. Ourselin, R. Mohiaddin, J. Keegan, D. Firmin, and G. Yang. Evaluation of algorithms for multi-modality whole heart segmentation: an open-access grand challenge. *Medical Image Analysis*, Vol. 58, 2019.
- [118] A. Mortazi, J. Burt, and U. Bagci. Multi-planar deep segmentation networks for cardiac substructures from MRI and CT. In *International Workshop on Statistical Atlases and Computational Models of the Heart (Springer)*, pp. 199–206, 2017.
- [119] Q. Tong, M. Ning, W. Si, X. Liao, and J. Qin. 3D deeply-supervised U-net based whole heart segmentation. In *International Workshop on Statistical Atlases and Computational Models of the Heart (Springer)*, pp. 224–232, 2017.
- [120] C. Wang and O. Smedby. Automatic whole heart segmentation using deep learning and shape context. *ACDC and MMWHS Challenges, 8th International Workshop, STACOM 2017, Held in Conjunction with MICCAI 2017, Revised Selected Papers (Springer)*, pp. 242–249, Quebec City, Canada, Sep. 2017.
- [121] C. Payer, D. Stern, H. Bischof, and M. Urschler. Multi-label whole heart segmentation using

- cnns and anatomical label configurations. In *International Workshop on Statistical Atlases and Computational Models of the Heart* (Springer International Publishing), pp. 190–198, 2018.
- [122] X. Yang, C. Bian, L. Yu, D. Ni, and P. Heng. 3D convolutional networks for fully automatic fine-grained whole heart partition. In *International Workshop on Statistical Atlases and Computational Models of the Heart* (Springer), pp. 181–189, 2017.
- [123] M. Zreik, T. Leiner, B. D. D. Vos, R. W. V. Hamersvelt, M. A. Viergever and I. Išgum. Automatic segmentation of the left ventricle in cardiac CT angiography using convolutional neural networks. 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI), pp. 40-43. 2016.
- [124] B. D. D. Vos, J. M. Wolterink, P. A. D. Jong, T. Leiner, M. A. Viergever, and I. Isgum. Convnet-based localization of anatomical structures in 3-D medical images. *IEEE Transactions on Medical Imaging*, vol. 36, pp. 1470–1481, 2017.
- [125] Z. Xu, Z. Wu, and J. Feng. Cfun: combining faster R-cnn and U-net network for efficient whole heart segmentation. *arXiv preprint arXiv: 1812.04914*, 2019.
- [126] G. Carneiro and J. C. Nascimento. Combining multiple dynamic models and deep learning architectures for tracking the left ventricle endocardium in ultrasound data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 2592–2607, 2013.
- [127] G. Veni, M. Moradi, H. Bulu, G. Narayan and T. Syeda-Mahmood. Echocardiography segmentation based on a shape-guided deformable model driven by a fully convolutional network prior. 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), pp. 898-902, 2018.
- [128] G. Carneiro and J. C. Nascimento. Multiple dynamic models for tracking the left ventricle of the heart from ultrasound data using particle filters and deep learning architectures. In *Conference on Computer Vision and Pattern Recognition (IEEE)*, pp. 2815–2822, 2010.
- [129] G. Carneiro, J. Nascimento, and A. Freitas. Robust left ventricle segmentation from ultrasound data using deep neural networks and efficient search methods. In *2010 IEEE International Symposium on Biomedical Imaging: from Nano to Macro*. pp. 1085–1088, 2010.

- [130] G. Carneiro, J. C. Nascimento, and A. Freitas. The segmentation of the left ventricle of the heart from ultrasound data using deep learning architectures and derivative-based search methods. *IEEE Transactions on Image Processing*, vol. 21, pp. 968–982, 2012.
- [131] E. Smistad, Andreas Østvik, Ivar Mjal Salte, S. Leclerc, O. Bernard, and L. Løvstakken. Fully automatic real-time ejection fraction and MAPSE measurements in 2D echocardiography using deep neural networks. *IEEE International Ultrasonics Symposium (IUS) (IEEE)*, pp. 1–4, 2018.
- [132] S. Leclerc, E. Smistad, T. Grenier, C. Lartizien, A. Ostvik, F. Espinosa, P. Jodoin, L. Lovstakken, and O. Bernard. Deep learning applied to multi-structure segmentation in 2D echocardiography: A preliminary investigation of the required database size. In *2018 IEEE International Ultrasonics Symposium (IUS) (IEEE)*, pp. 1–4, 2018.
- [133] O. Oktay, E. Ferrante, K. Kamnitsas, M. Heinrich, W. Bai, J. Caballero, S. Cook, A. D. Marvao, T. Dawes, D. O’Regan, B. Kainz, B. Glocker, and D. Rueckert. Anatomically constrained neural networks (Acnns): application to cardiac image enhancement and segmentation. *IEEE Transactions on Medical Imaging*, vol. 37, pp. 384–395, 2018.
- [134] S. Dong, G. Luo, k. Wang, S. Cao, A. Mercado, O. Shmulovich, H. Zhang, and S. Li. Voxe-lAtlasGAN: 3D left ventricle segmentation on echocardiography with atlas guided generation and voxel-to-voxel discrimination. In *Medical Image Computing and Computer Assisted Intervention – MICCAI (Springer International Publishing)*, pp. 622– 629, 2018.
- [135] K. Ta, S. S. Ahn, J. C. Stendahl, A. J. Sinusas and J. S. Duncan. Shape-regularized unsupervised left ventricular motion network with segmentation capability in 3D+ time echocardiography. *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 536-540, 2021.
- [136] P. Wang, Y. Gan, P. Shui, F. Yu, Y. Zhang, S. Chen, and Z. Sun. 3D shape segmentation via shape fully convolutional networks. *Computers Graphics*, vol. 70, pp. 128–139, 2018.
- [137] L. Sun, S. Zhang, H. Chen, and L. Luo. Cardiac tumor segmentation and survival prediction using multimodal MRI scans with deep learning. *Frontiers in Neuroscience*, vol. 13, Aug. 2019.

- [138] S. Valipour, M. Siam, M. Jagersand, and N. Ray. Recurrent fully convolutional networks for video segmentation. arXiv preprint arXiv:1606.00487, 2016.
- [139] M. Fayyaz, M. H. Saffar, M. Sabokrou, M. Fathy, R. Klette, and F. Huang. Stfcn: spatio-temporal fcn for semantic video segmentation. arXiv preprint arXiv: 1608.05971, 2016.
- [140] J. Chen, L. Yang, Y. Zhang, M. Alber, and D. Z. Chen. Combining fully convolutional and recurrent neural networks for 3D biomedical image segmentation. In *Advances in Neural Information Processing Systems*, pp. 3036–3044, 2016.
- [141] C. Dahmen, K. Belharet, D. Folio, A. Ferreira, and S. Fatikow. MRI-based dynamic tracking of an untethered ferromagnetic microcapsule navigating in liquid. *International Journal of Optomechatronics*, vol. 10, pp. 73-96, 2016.
- [142] Z. Li, C. Li, L. Dong, and J. Zhao. A review of microrobot’s system: towards system integration for autonomous actuation in vivo. *Micromachines (Basel)*, 12(10):1249, 2021.
- [143] S. Martel, O. Felfoul, J. Mathieu, A. Chanu, S. Tamaz, M. Mohammadi, M. Mankiewicz, and N. Tabatabaei. MRI-based medical nanorobotic platform for the control of magnetic nanoparticles and flagellated bacteria for target interventions in human capillaries. *The International Journal of Robotics Research*, vol. 28, pp. 1169-1182, 2009.
- [144] K. Belharet, D. Folio and A. Ferreira. Endovascular navigation of a ferromagnetic microrobot using MRI-based predictive control. *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2804-2809, 2010.
- [145] S. Martel. Swimming microorganisms acting as nanorobots versus artificial nanorobotic agents: a perspective view from an historical retrospective on the future of medical nanorobotics in the largest known three-dimensional biomicrofluidic networks. *Biomicrofluidics*, vol. 10, 2 021301, 2016.
- [146] D. Huh, J. Park, U. Huh, and H. Kim. Path planning and navigation for autonomous mobile robot. *IEEE 2002 28th Annual Conference of the Industrial Electronics Society*, vol. 2, pp. 1538-1542, 2002.

- [147] K. Belharet, D. Folio and A. Ferreira. 3D MRI-based predictive control of a ferromagnetic microrobot navigating in blood vessels. 2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics, pp. 808-813, 2010.
- [148] S. N. Tabatabaei, H. Girouard, A. Carret, S. Martel. Toward nonsystemic delivery of therapeutics across the blood-cardiac barrier. *Nanomedicine (Lond)*, 10(14): 2129-31, 2015.
- [149] S. Martel. Microrobotic navigable entities for magnetic resonance targeting. Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 1942–1945, 2010.
- [150] S. N. Tabatabaei, S. Duchemin, H. Girouard, and S. Martel. Towards mr-navigable nanorobotic carriers for drug delivery into the cardiac. *IEEE International Conference on Robotics and Automation*, pp. 727-732, 2012.
- [151] S. Tamaz, R. Gourdeau, A. Chanu, J. B. Mathieu and S. Martel. Real-time MRI-based control of a ferromagnetic core for endovascular navigation. In *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 7, pp. 1854-1863, July 2008.
- [152] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
- [153] T. Küstner, N. Fuin, K. Hammernik, A. Bustin, H. Qi, R. Hajhosseiny, P. G. Masci, R. Neji, D. Rueckert, R. M. Botnar, and C. Prieto. CINENet: deep learning-based 3D cardiac CINE MRI reconstruction with multi-coil complex-valued 4D spatio-temporal convolutions. *Scientific Reports*, vol. 10, no. 1, 2020.