Towards Practical Ultrasound AI Across Real-World Patient Diversity

Edward Chen

CMU-RI

August 2021

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
John Galeotti, Chair
Howie Choset, Chair
Deva Ramanan
Chih-Kuan Yeh

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

Abstract

Needle-puncture procedures are often used to treat patients with traumatic and life-threatening injuries. However, properly locating the safest needle insertion location, such as the femoral region, in such high-tempo situations is difficult and can lead to severe complications. The aim of this thesis is to address this difficulty by developing an automatic robot-guided needle insertion system. To close the loop, it requires an imaging modality so the ultrasound modality was used due to its lack of radiation, low costs, and portability.

A major cause for the majority of complications related to needle-puncture procedures is human judgment. In an attempt to minimize the amount of human input for such procedures, which may be clouded by emotions, this thesis aims to fully automate the procedure. As deep neural networks are capable of learning more complex, non-linear functions to approximate the data, and have the potential to generalize well given sufficient data, this thesis leverages the power of deep neural networks and computer vision. Because localization of the proper anatomical landmarks is critical for percutaneous (needle-puncture) procedures, this work focuses on the task of semantic segmentation - which aims to classify each pixel of the images. However, ultrasound images present their own set of challenges: (1) extremely noisy images, which often necessitates trained medical professionals for interpretation, resulting in training data being expensive to collect, and (2) immense variations across ultrasound scanners, imaging settings, body types, and injury scenarios. I aim to address such challenges in the four works included in this thesis.

In the first part of this thesis, I present a deeper introduction of the ultrasound imaging challenges we face as well as a short background of the imaging modality. I then continue with work on studying how semantic segmentation networks can generalize across different populations of ultrasound images using a technique known as transfer learning. The second work then more directly addresses the high-costs of training data for ultrasound images. The proposed method introduces novel temporal data augmentation strategies to increase the size of training data, specifically for dealing with various ultrasound scanning patterns. I evaluate our methods on multiple types of scanning patterns and notice improvements with our simple stochastic augmentation methods. The following work focuses more on addressing the variations across body and injury types when imaging them. This thesis introduces a novel spatial non-uniform data augmentation method which is able to deform various sections of the ultrasound images to mimic long-tailed scenarios.

The final portion of this thesis introduces an initial prototype for a robotic system to automatically insert a needle into the femoral region of a patient. This prototype only represents the first step in achieving our long-term goal; the system introduced aims to determine the safest insertion point for the needle. I believe there is a significant amount more which can be built on top of all these works described and plan to pursue such further in the future.

Acknowledgments

I would first like to thank my advisors, Professors John Galeotti and Howie Choset, for their remarkable support over my past two years at Carnegie Mellon University. I am eternally grateful for the feedback, stories, lessons, and advice which both of my advisors have graciously provided to me. I am also extremely grateful for the members of both the BIG Lab and Biorobotics Lab for providing me with valuable experiences, companionship, and feedback throughout my time here. Many of my lab mates have been extremely instrumental in me achieving the research I have output. I would like to specifically thank members of my research team such as Abhimanyu, Alex Hung, Evan Harber, Gautam Gare, Nico Zevallos, Wanwen Chen, and others for being such excellent researchers and research companions.

Additionally, I would like to sincerely thank the other members of my thesis committee, Professor Deva Ramanan and Chih-Kuan Yeh. I am extremely grateful for both of their support throughout this long process. They have both provided me extremely valuable advice and suggestions for furthering my research to the next level.

Finally, I would like to thank my parents, brother, and friends for their consistent support throughout all of these years. My parents sacrificed many of their dreams, friends, and comfort in their own country to come to a foreign land and support my brother and I so we can live better lives. I am forever indebted to their gratitude and care.

Contents

1 Introduction 1

1.1 Background Information 3

1.2 Datasets 3

2 A Study of Domain Generalization on Ultrasound-based Multi-Class Segmentation of Arteries, Veins, Ligaments, and Nerves Using Transfer Learning 7

2.1 Introduction 7

2.2 Materials And Methods 9

2.3 Experiments And Results 10

2.3.1 Encoder vs. Decoder *OOTD* Paired-Difference Statistical Significance Testing 11

2.3.2 Contiguous Encoder/Decoder Blocks vs. *OOTD* Scores Linear Relationship Statistical Significance Testing 12

2.3.3 Autoencoder Reconstruction Experiment Details 12

3 Stochastic Temporal Data Augmentation for Adaptation to Out-of-Distribution Temporal Features 15

3.1 Introduction 15

3.2 Methods 17

3.3 Experiments 19

3.4 Results and Discussion 21

3.5 Conclusion and Future Work 23

4 Uncertainty-based Adaptive Data Augmentation for Ultrasound Anatomical Variations 25

4.1 Introduction 25

4.2 Methods 27

4.2.1 Semantic Segmentation Neural Network 27

4.2.2 Augmentation Module 28

4.3 Experiments and Results 30

4.3.1 Data 30

4.3.2 Training Details 30

4.3.3 Experiments 30

4.3.4 Metrics 30

4.4 Discussion 32

4.5 Conclusion and Future Work 33

5 Multi-Class Bayesian Segmentation of Robotically Acquired Ultrasound Enabling 3D Site Selection along Femoral Vessels for Planning Safer Needle Insertion 35

5.1 Introduction 35

5.2 Related work 38

5.3 Methods 40

5.3.1 Experimental Setup 40

5.3.2 Robot Controller 40

5.3.3 Multi-Class Segmentation of Arteries, Veins, Ligaments, and Nerves 41

5.3.4 3D Visualization of Multi-Class Segmentation 41

5.3.5 Optimal Needle Insertion Planning Algorithm 42

5.3.6 Fully Automatic Pipeline 45

5.4 Analysis and Results 46

5.4.1 Robot Controller 46

5.4.2 Multi-Class Segmentation 47

5.4.3 Optimal Needle Insertion Planning 48

5.5 Conclusion and Future Work 51

6 Conclusion 53

Bibliography 55

List of Figures

1.1 Ultrasound scan depiction across human hip-and-thigh (femoral) region. US refers to the ultrasound probe, and t_i refers to the different timestamps, in order. 4

1.2 Ultrasound image axes, shown for a linear-probe image. 4

2.1 Sample images of (from the left): *h50*, *ph1-12*, *ph1-22*, *ph2-12*, *p12* 9

2.2 Our U-Net naming convention 10

2.3 Best-fit lines showing positive relationship between longer encoder subsequences and *OOTD* scores for each of the 4 experiments 11

2.4 Best-fit lines showing positive relationship between longer decoder subsequences and *OOTD* scores, except for case 2 which may be affected by class count 11

3.1 Sample images from all of the data categories used in this study. Starting from left: *standard-POCUS*, *sudden-changes*, *right-side*, *higher-depth*. **Color Key**: Red: Artery, Green: Ligament, Blue: Vein, Yellow: Nerve. . . 19

3.2 3D U-Net architecture. The encoder blocks (blue) consists of the input layer followed by the 2 pairs of convolutional layer, batch normalization, and ReLu activation. The decoder blocks (red) consists of the same structure. The down and up arrows represent downsampling and upsampling layers. 20

4.1 Flow diagram of the overall uncertainty-based augmentation pipeline. The uncertainty maps from the segmentation network get passed into the agent neural network, which then generates the control points for deforming the image in the augmentation module. Those are then fed back into the training process. 27

4.2 Example ultrasound image undergoing the image deformation. The top images visualize the original and new images, whereas the bottom images visualize the same images' labels. 33

5.1 Portable robotic system designed for ultrasound-guided needle site selection. The image on the left displays a sample set-up of the pipeline. The image on the right displays the final 3D visualization with the optimal insertion point (white) for the left side of a torso phantom. **Color Key**: Artery - Red, Vein - Blue, Ligament - Green, Nerve - Yellow. 36

5.2 Flow diagram of the overall automatic pipeline. The robot system first collects ultrasound images from its scanning process, sends them to the deep learning model for segmentation, retrieves the segmentation coordinates for 3D model generation, and then outputs suggested locations for needle insertion based on safety standards. 36

5.3 Diagram of our Bayesian 3D U-Net model. The blocks represent 3D convolutions + Batch Normalization + ReLu, the dotted lines represent dropout, and the final outputs represent the predictive mean and variance. 42

5.4 Diagram of ideal femoral arterial puncture site [1]. The 3 cm range shown in the figure indicates ideal region for needle insertion into femoral artery. Used Creative Commons Attribution Non-Commercial License. 43

5.5 Examples of the variety in images for which we validated our methods on, with the labels overlaid on top. From left to right: *blue-gel*, *torso-left*, *torso-right*, *live-pig*. Each row shows the 3 different imaging settings with which we varied for each subject, except for *live-pig*. **Color Key**: Arteries - Red, Veins - Blue, Ligaments - Green, Nerves - Yellow. 44

5.6 Boxplots displaying the mean and standard deviation of the force response values with the force constraint on (left) and off (right). This illustrates the stability of our contact force due to the force constraint. 46

5.7 Our adaptive NCC score for different ultrasound scanning scenarios. For all of our trials, the stability of the scanned ultrasound images approached the upper bound of a smooth scan. 47

5.8 3D visualization of *blue-gel*. The optimal insertion point is the white dot, whereas the ligament and arterial bifurcation points are the grey dots. The image on the bottom illustrates a heatmap of the *Total Site Scores* for the needle planning algorithm (unsafe to safe goes from gray to red). 48

5.9 Segmentation results on images collected with different imaging settings and anatomy. Left to right: *blue-gel*, *torso-left*, *torso-right*, *live-pig*. 48

5.10 3D visualization of *torso-right*. Optimal insertion point is the white dot, whereas the ligament and arterial bifurcation points are the grey dots. The image on the bottom illustrates a heatmap of the *Total Site Scores* for the needle planning algorithm (unsafe to safe goes from gray to red). 49

5.11 3D visualization of *live-pig*. The top image shows the method we use for detecting the ligament. The optimal insertion point is the white dot, whereas the ligament and arterial bifurcation points are the grey dots. The image on the bottom illustrates a heatmap of the *Total Site Scores* for the needle planning algorithm (unsafe to safe goes from gray to red). 49

List of Tables

2.3 Transfer Learning Experimental Results using Dice Coefficient metric (averaged over 3 runs) comparing across changes in anatomy and imaging settings. Encoder 1 (e-1) means that only the first block in the encoder is used for fine-tuning, encoder 2 (e-2) means 2 blocks in the encoder are used, decoder 4 (d-4) refers to only the first block on the decoder side, and so on. *OOTD* is the arithmetic average of the Dice metric on the unseen datasets. 13

2.4 Transfer Learning Experimental Results using Dice Coefficient metric (averaged over 3 runs) comparing across imaging settings. Encoder 1 (e-1) means that only the first block in the encoder is used for fine-tuning, encoder 2 (e-2) means 2 blocks in the encoder are used, decoder 4 (d-4) refers to only the first block on the decoder side, and so on. *OOTD* is the arithmetic average of the Dice metric on the unseen datasets. 14

3.1 Region similarity comparing the best across all ablations, on a dataset in the same domain, *standard-POCUS*, and a dataset with temporal shifts, *sudden-changes*. Bolded values represent our methods which surpassed the results from the baselines. 21

3.2 Contour accuracy comparing the best across all ablations, on a dataset in the same domain, *standard-POCUS*, and a dataset with temporal shifts, *sudden-changes*. Bolded values represent our methods which surpassed the results from the baselines. 21

3.3 Region similarity comparing the best across all ablations, on datasets outside of training domain. Bolded values represent our methods which surpassed the results from the baselines. 21

4.1 Region similarity metric comparing the best-performing methods, across the 2 datasets. The bolded values represent our methods which surpassed the baseline results. 31

5.1 Evaluation with region similarity (J), contour accuracy (F), and temporal stability (T). Arrows indicate optimal direction for avg ± std across im_a, im_b, im_c (im_a for *live-pig*). 50

Chapter 1

Introduction

Needle-puncture procedures are one method commonly used by clinicians to perform life-saving and trauma-stabilizing interventions, such as providing intravenous medicine and reducing aortic blood flow to reduce massive internal hemorrhaging. However, during such traumatic situations, clinicians are often pressured for time or lethargic from a string of medical cases, thereby increasing the likelihood of complications associated with the procedures. Robot-guided catheter insertion is one method which can be used to avoid any mistakes while still delivering a similar quality of medical care to the patient. To be able to automatically determine the proper needle insertion location, though, a suitable medical imaging modality is necessary. Although there are several available options, this work uses ultrasound (US) imaging due to its portability, safety, and low-costs. Throughout this work, ultrasound imaging is used for robot-guided needle insertion procedures within the femoral region, which allows insertion of larger arterial and venous catheters than any other location.

Identifying landmarks in the femoral area is crucial for US-based robot-guided catheter insertion, however the variability of particular US scanner and settings used, per-patient differences, and unique situations can lead to a variety of potential issues. The ultrasound image presentations may vary when imaged with different scanners, on different patient body types, in various injury scenarios, and with different ultrasound scanning rates. Along with this, the inherent noisiness of ultrasound images leads to an often-small amount of labeled training data as they often require medical expertise to label, making them very expensive to collect. As such, the wide variety in presentation and dearth of training data often caused the performance of past deep learning-based approaches to be narrowly limited to the training data distribution. These limitations, however, can be (at least partially) alleviated by various methods such as transfer learning and data augmentation.

The first part of this thesis studies the problem of generalizing across different ultrasound imaging scenarios through the lens of transfer learning, a machine learning method in which the weights of a prior trained model are reused for a new task. By reusing previously trained models, we can leverage some of the more general learned features for our new task without requiring the additional data to relearn them. We note that the often severe lack of ultrasound image training data can be circumvented by fine-tuning all or part of the model, yet the effects of fine-tuning are seldom discussed. This work studies the US-based segmentation of multiple classes through transfer learning by fine-tuning different contiguous blocks within the model, and evaluating on a gamut of US data from different scanners and settings. The work proposes a simple method for estimating which fine-tuning method results in higher generalization on unseen datasets and observe statistically significant differences between the fine-tuning methods while working towards domain generalization.

The second part of this thesis focuses on addressing the lack of data in ultrasound imaging when dealing with generalizing across different ultrasound scanning patterns - in the temporal sense. The method proposed aims to improve temporal generalization using temporal data augmentation. Traditional image data augmentation methods consist of simple spatial transformations such as crops, rotations, and translations. However, such methods do not address the temporal features in the often erratic vessel pulsations and ultrasound scanning methods during emergency scenarios. This section aims to explicitly improve the model's robustness and generalization to various scanning patterns, which can each be viewed as a separate temporal domain, and propose several novel stochastic temporal augmentation strategies to address the variability in scanning by including otherwise out-of-(temporal)-domain samples within the augmentation. The work contains experiments with various novel spatial-temporal data augmentation approaches, which instill temporal shifts into sequences that originally did not contain them. The proposed methods performed better than current methods on 7/8 trials on an unseen, out-of-domain dataset collected with erratic scanning and on an unseen, original-domain dataset.

The third part of this thesis addresses the lack of ultrasound imaging data from the perspective of the various injury scenarios and body types which are present in real-world imaging scenarios. The work aims to improve spatial generalization with a novel method for data augmentation. Overall, data augmentation remains to be a simple and inexpensive method for generalizing across unseen domains. Current data augmentation methods for ultrasound imaging involve simple image transformations - rotations, flips, skews, and blurs - but are not able to adapt to the current state of the deep learning model. This work presents the first online adaptive data augmentation method that is able to generate synthetic training data on-the-fly, enabling the model to adapt to countless spatial deformations. The proposed method leverages prior work on uncertainty quantification to understand the model's weaknesses at any given stage. The method is also able to then "spot-augment" subsets of regions within the ultrasound image, all in a real-time manner. The work also shows that the proposed method is able to perform significantly better in out-of-training distributions, when compared against models trained on the same dataset.

The final part of this thesis introduces an initial prototype of a robot-guided catheter insertion system by presenting the first known robotic system that creates accurate maps of many anatomic structures in the femoral region using ultrasound information. This map comes in the form of a three-dimensional point cloud where points are properly labeled, e.g., veins, ligaments, arteries, etc. A multi-class, multi-instance Bayesian 3D convolutional neural network (CNN) is used to segment and identify the anatomic structures from 2D time series ultrasound data. The 2D results are then combined with each other and the kinematics of the robot that is moving the ultrasound probe to create a 3D point cloud. This 3D point cloud is then analyzed, based on standard-clinical-practice heuristic rules, to determine an ideal point, in 3D space, to puncture with the needle. In particular, the algorithm determines for the desired point in either the common femoral artery or vein. Once the patient is within the robotic workspace, the steps from ultrasound scanning to needle insertion planning can be completed with minimal human intervention and are designed to be robust to various situations - such as changes due to ultrasound imaging settings and anatomical variations.

1.1 Background Information

The use of ultrasound imaging for medicine dates back more than 50 years [2]. Ultrasound imaging has since then evolved tremendously and is often lauded for its portability, safety, and low-costs, which are the reasons for why it was chosen. However, ultrasound imaging does bring with it a major challenges: noisy images with substantial differences across ultrasound scanners and imaging settings.

The primary differences in ultrasound scanners and imaging settings in the works in this thesis can be largely attributed to the ultrasound frequency, depth of imaging, and per-depth gain values. Ultrasound scanners produce images by emitting sound waves from an acoustic transducer inside a hand-held probe into the patient's body. (The terms "transducer" and "probe" are often used interchangeably.) The number of vibration cycles within a second for the emitted waves is known as the ultrasound frequency; this can vary by ultrasound probe, typically ranging from 2 - 30 MHz. The depth value is a setting which can be adjusted on the ultrasound scanner and measures how far below the patient's skin the scanner can image; it is also inversely related to the ultrasound frequency, since higher frequencies (which resolve more detail) are more rapidly attenuated by tissue. The ultrasound gain values include both an overall amplification of ultrasound waves as well as depth-specific additional amplification, essentially giving the image a brighter or darker look overall and at various depths[2]; when a single gain setting is discussed, it may be inferred to refer to the overall gain.

In relation to the rest of this thesis, I would like to emphasize that the width of the ultrasound image produced corresponds to the width of the linear ultrasound probe or to the angle (and depth) of the curvilinear or phased-array ultrasound probe, and the height of the ultrasound image corresponds to the depth of the ultrasound waves. (This work uses a linear probe.) Such ultrasound images are often collected in continuous scans around the patient's body. Each ultrasound scan can contain hundreds of ultrasound frames. I note that, in some of the works which follow, I treat the ultrasound frames as individual images when input to the networks and in other cases we pass as input small sequences of ultrasound images, thereby exhibiting some temporal features. Images 1.1 and 1.2 below illustrate this concept further.

1.2 Datasets

The works in this thesis used datasets which attempted to obtain training samples from a diverse subset of the real-world distribution of vascular and emergency ultrasound imaging. In total, this thesis includes 6 datasets, with each dataset containing multiple video sequences. Data was acquired from a phantom, a preexisting de-identified human subject [3], and live-pig subjects. The phantom that was used in this work was the CAE Blue Phan-

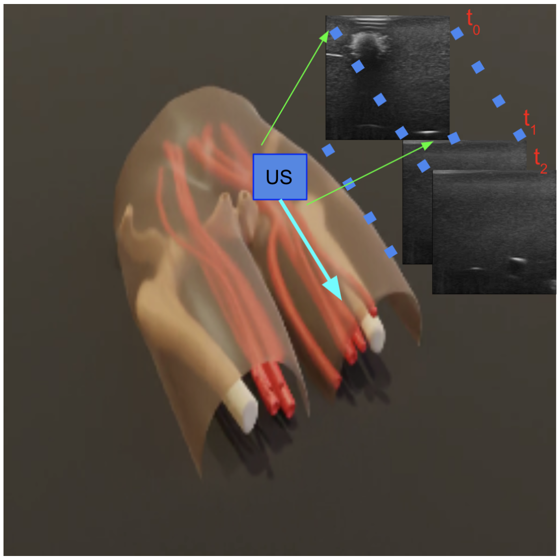


Figure 1.1: Ultrasound scan depiction across human hip-and-thigh (femoral) region. US refers to the ultrasound probe, and t_i refers to the different timestamps, in order.

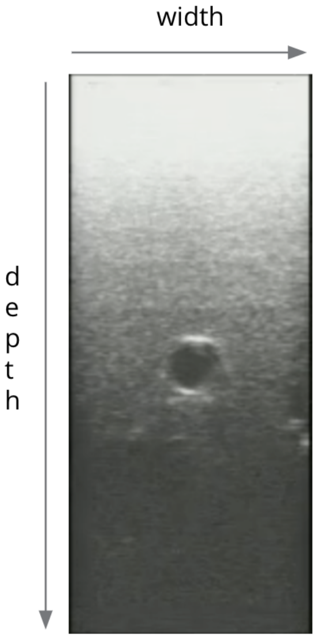


Figure 1.2: Ultrasound image axes, shown for a linear-probe image.

tom femoral vascular access lower torso ultrasound training model (BPF1500-HP). In the human subjects, the arteries and veins in the palmar arch of the hand were imaged.

Scanners: Three different ultrasound scanners were used for imaging the phantom, pig, and human subjects: (1) Fukuda Denshi portable (i.e. Point of Care Ultrasound, *POCUS*) scanner with a 5-12 MHz linear transducer, (2) Diasus High-Frequency Ultrasound (*HFUS*) scanner (Dynamic Imaging, UK) with a 10-22 MHz linear transducer, and (3) a VisualSonics Vevo 2100 Ultra High Frequency Ultrasound (*UHFUS*) scanner with a 50MHz linear transducer. The images include diverse scan parameters and settings (e.g. gain values) and anatomical variations. Datasets representing this diversity of imaging settings include: *human-single50*, from the UHFUS machine, *phantom1-multi22*, from the HFUS machine, and *phantom1-multi12*, from the lower frequency Fukuda Denshi machine. The numerical suffix for each dataset name represents the ultrasound frequency with which it was collected.

Phantom-Based Categories: Three categories of imaged sequences were acquired from the phantom: (1) *phantom1-multi12*, (2) *phantom1-multi22*, and (3) *phantom2-multi12*. The prefix "phantom1" represents image sequences collected from the left side of the phantom. On the other hand, the prefix "phantom2" represents data collected from the right side of the phantom, which was manufactured differently to provide diversity for human training, and as such contains different anatomy (muscles, liver, etc.) and other artifacts unknown to an AI model trained only on the left side, resulting in a more complex dataset. Each of these datasets consisted of the following classes: arteries, veins, ligaments, and nerves.

Scanning-Pattern-Based Categories While most of the datasets were collected with a smooth, uniform scanning pattern, I also collected a single dataset, *sudden-changes*, which involved extensive probe motions, such as fast, irregular, and erratic probe motion, and out-of-plane deformation. The dataset *sudden-changes* still contains the same classes: arteries, veins, ligaments, and nerves. As a result of the erratic probe motion, potentially confounding motion-related imaging artifacts were also present in *sudden-changes*.

Human Subject-Based Category: The arteries and veins in the palmar arch of the hand [4] were imaged using the UHFUS (50 MHz) scanner, and a wide range of overall and per-depth gain values were used (40-70 dB). Each sequence consisted of only one artery or vein, and the same expert labeled the vessel in each frame (herein assumed to be an artery for training purposes). I refer to this UHFUS human category as *human-single50*.

Live Pig-Based Category: The femoral arteries and veins of a live pig were imaged using the Fukuda Denshi portable scanner with a 5-12 MHz transducer. Arteries and veins were separately labeled as such. I refer to this category as *pig-multi12*.

Substantial Noise and Artifacts: Of the 5 datasets, *human-single50* consists of the most amount of noise and speckle as a result of its high frequency, then closely followed by *phantom1-multi22* for the same reason. *phantom2-multi12*, on the other hand, contains more artifacts as a result of the extra anatomic structures located at that position on the phantom.

Data Quantity: The *phantom1-multi12* category consisted of 12 sequences, each containing 50 frames totalling 600 frames. The *phantom1-multi22* category consisted of 18 sequences, each containing 31 frames totalling 558 frames. The *phantom2-multi12* category consisted of 3 sequences, with each sequence containing 50 frames totalling 150 frames.

For the *human-single50* category, 10 sequences were obtained in total from the left and right hands of 4 subjects [3], with each sequence containing 50 frames totalling 500 frames. Lastly, the *pig-multi12* category consisted of 3 sequences, with each sequence containing 40 frames totalling 120 frames. All classes (arteries, veins, ligaments and nerves) were present in (at least some frames of) every sequence in *phantom1-multi-12*, *phantom1-multi22*, and *phantom2-multi12*. Only a single vessel class (of either artery or vein) was present in *human-single50* whereas both artery and vein classes were present in *pig-multi12*. Each frame was annotated by a single expert, from a diverse pool of experts used across these datasets.

Chapter 2

A Study of Domain Generalization on Ultrasound-based Multi-Class Segmentation of Arteries, Veins, Ligaments, and Nerves Using Transfer Learning

Chapter 2 is adapted from the publication:
Edward Chen, Tejas Sudharshan Mathai, Vinit Sarode, Howie Choset, and John Galeotti, "A Study of Domain Generalization on Ultrasound-based Multi-Class Segmentation of Arteries, Veins, Ligaments, and Nerves Using Transfer Learning," NeurIPS Machine Learning for Health Workshop (ML4H), 2020
Edward Chen's contributions as the first author of the publication include: conducting initial literature review, developing the algorithm and experiments, conducting the experiments, analyzing the data, writing the manuscripts and responding to reviewers with revisions. Tejas Sudharshan Mathai and Vinit Sarode assisted with providing experimental suggestions and running actual experiments. Howie Choset and John Galeotti are the supervising faculty advisors.

2.1 Introduction

In the case of major internal hemorrhaging, real-time ultrasound (US) imaging can guide the robotic insertion of a vascular catheter for Resuscitative Endovascular Balloon Occlusion of the Aorta (REBOA) via the femoral artery to prevent the patient from bleeding to death. Automatically segmenting femoral area landmarks will be crucial to the optimal catheter placement in time-sensitive situations. To this end, the developed technology has to be robust to variations in anatomy, scanner settings, external artifacts (in traumatic injury scenarios), probe positioning, etc. However, medical imaging datasets are often limited in quantity and span a restricted distribution over the data space [5]. Deep learning models trained on such data perform poorly when tested on data from different anatomic areas or scanner settings [6], thereby limiting their real-world usage. For instance, [7] illustrate the first-attempt success rate for ultrasound-guided needle insertions dropping by $\sim$30% on datasets of different anatomy and settings, leaving room for serious consequences to the patient [8].

The present standard for improving robustness in the medical domain is to use an existing architecture trained on natural images, such as ImageNet [9, 10], and then fine-tune on the medical images [11, 12]. However, little work has been done to illuminate the generalization ability of models to medical images using transfer learning, and to understand segmentation models commonly used in medical robotics [13], such as the U-Net [14]. Raghu, et. al. [11] performed an in-depth study of transfer learning for classification of medical images, albeit starting from natural images. Amiri, et. al. [15] studied UNet-based fine-tuning, but on a single domain.

We aim to expand the current understanding of domain generalization and provide actionable insights for enhanced robustness within the context of ultrasound-based multi-class segmentation using transfer learning. Here, we consider the practical case where the data is gathered in a sequential manner, specifically when the previously trained data is unavailable due to privacy restrictions [16]. In reality, deep learning models for medical devices will also often have been trained on some subset of medical imaging domains, for which labelled data was available. We further design the experiments in a way to more explicitly control for certain training domains, attempting to generalize training across more of the real-world space of unexpected images.

From our experiments, we reveal the following insights: (1) As consecutive blocks on both the encoder and decoder side are individually fine-tuned, the out-of-training-domain (*OOTD*) performance generally increases. The *OOTD* data are different from the pre-training data (*pt-data*) and fine-tuning data (*ft-data*). (2) Having a smaller number of classes in the *pt-data* may hamper the final performance on the *ft-data*, but not of that for the *OOTD* data, and (3) There is a statistically significant difference between fine-tuning the encoder and decoder in terms of performance on *OOTD* data. We then take into account such observations and propose selecting the *ft-data* performance as a proxy for OOTD performance when selecting the best fine-tuning method to use.

2.2 Materials And Methods

For our datasets, we use the same ones as described in Section 1.2.
Scanners: 3 different ultrasound scanners with diverse scan settings (e.g. gain values) were used: (1) a portable scanner, (2) a high-frequency ultrasound (*HFUS*) scanner, and (3) an ultra high-frequency ultrasound (*UHFUS*) scanner. More details in 1.2.
Human Data: The UHFUS scanner imaged arteries and veins in human subjects [4], and a single class label was assigned to them by an expert. We refer to this data as *human-single50* (*h50*).
Phantom Data: 3 categories of sequences were acquired from a phantom: (1) *phantom1-multi12* (*ph1-12*), (2) *phantom1-multi22* (*ph1-22*), and (3) *phantom2-multi12* (*ph2-12*). The prefix "phantom1" represents image sequences collected from the left side of the phantom, while prefix "phantom2" represents data collected from the right side of the phantom, which also contained different anatomy (muscles, liver, etc.). 4 classes were labelled in each phantom dataset: arteries, veins, ligaments, and nerves.
Pig Data: Data was gathered from a living pig using the portable scanner, and the arteries and veins were labelled as 2 classes. We refer to this as *pig-multi12* (*p12*). Each numerical suffix represents the frequency with which it was collected with. More details in 1.2.

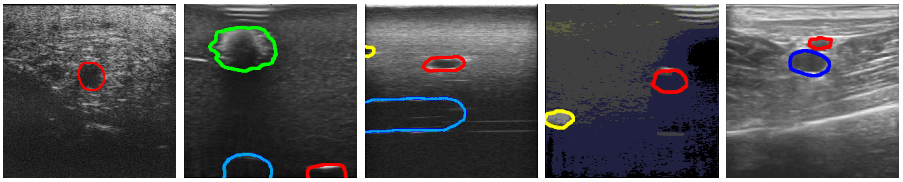


Figure 2.1: Sample images of (from the left): *h50*, *ph1-12*, *ph1-22*, *ph2-12*, *p12*

Transfer Learning: We evaluated segmentation performance on a U-Net model consisting of 5 encoder blocks (including the bottleneck layer as the 5th encoder) and 4 decoder blocks. We first transferred previously learned weights and then fine-tuned different models that spanned various **contiguous** blocks of the architecture. The encoder blocks were numbered from 1 to 5, starting from the input layer and ending with the bottleneck layer. The numbering for the encoder side is cumulative, e.g. "Encoder 5" refers to all 5 of the blocks leading up to the bottleneck layer. The decoder blocks were numbered from 4 (just after the bottleneck) to 1 for the output layer. "Decoder 1" in this case refers to the 4 blocks of the decoder up to the output layer. Figure 2.2 illustrates this naming convention.
Training Details: For each of the encoder and decoder transfer learning scenarios, we set batch normalization to use the overall training data's statistics, as opposed to batch statistics, as that is often what is used in practice.
Each of the training datasets consisted of close to 600 training images each. Each of the datasets, including *pig-multi12 (p12)* and *phantom2-multi12 (ph2-12)*, contains 150 images for testing. Similar to [3], we trained the multi-class segmentation models by resizing each ultrasound B-scan to 256x256 pixels. Traditional (spatial) data augmentations were done by random flipping, rotating, blurring, and translating the training set, such that each experimental run's training set was increased to $\sim$ 12,000 images. All experiments were conducted using TensorFlow [17], training with the Adam optimizer [18] on cross-entropy

loss with a batch size of 16, learning rate of 0.0001 for pre-training, and learning rate of 0.000001 for fine-tuning. Final pixel-level probabilities were classified using softmax, and the results were evaluated using the Dice Similarity Coefficient (DSC).

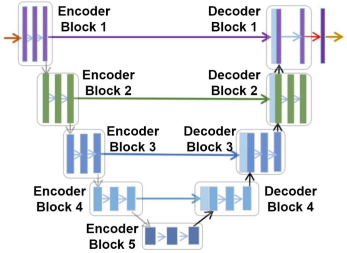


Figure 2.2: Our U-Net naming convention

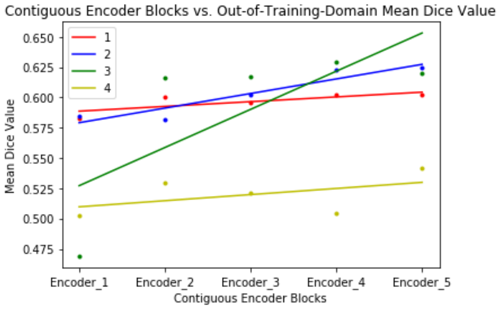
2.3 Experiments And Results

Discussion: For all 4 of the experiments, we noted a general trend: (1) as a larger number of the encoder/decoder blocks are fine-tuned, the model performed equally well or better on data from both its *ft-data* domain and *OOTD* data. We visualize this pattern in Figures 2.3 and 2.4. We describe significance testing details later. We note that the one exception to this is the decoder branch of experiment 2 (Figure 2.4), which leads to our next observation: (2) for models in experiment 2, we noticed that it was "difficult" for the batch normalization statistics to converge during the fine-tuning process. We believe this to be due to the fewer number of classes in the *pt-data* domain, leading to a more restrictive feature representation. On the contrary, the opposite is true for experiment 1, which has contiguous block-wise performances close to, and even surpassing, that of the full-model fine-tuning procedure. We attempt to further understand this class count-related effect together with another observation later in this section.

We additionally notice that (3) fine-tuning contiguous encoder blocks produced better *OOTD* performance than those with decoder blocks, which can also be seen in Tables 2.3 and 2.4 (2.3.1). This may be because fine-tuning blocks on the encoder side leads to a more diverse latent feature representation while retaining the localization information on the decoder side. We conducted Wilcoxon tests on the paired encoder-decoder differences for each of the 4 experiments. All had statistically significant greater *OOTD* performance from the encoder branch, except experiment 1, which may have had the decoder benefit from the greater number of classes in *pt-data*.

Considering the above, we propose to use each fine-tuning method's *ft-data* performance as a representation of its *OOTD* score. Our proposed method is to select the contiguous-block-wise fine-tuned model with the highest *ft-data* score. Note that the full-model fine-tuning always produced worse *OOTD* scores in our case. We can use such a method to predict a fine-tuned model that might have the best domain generalization capabilities while simultaneously selecting a fine-tuned model that performs well in its direct target task. We compare the *OOTD* performance of our method's model choices with that of the traditional full-model fine-tuning method in Table 2.5.

Figure 2.3: Best-fit lines showing positive relationship between longer encoder subsequences and *OOTD* scores for each of the 4 experiments



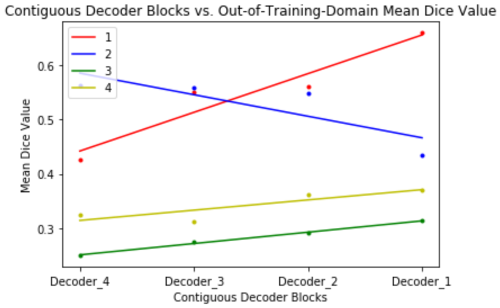


Figure 2.4: Best-fit lines showing positive relationship between longer decoder subsequences and *OOTD* scores, except for case 2 which may be affected by class count

In all cases, our method surpasses the *OOTD* performance with the full-model finetuning method by a large margin; the same occurs against those of decoder-only methods, which are also commonly used. Of note are experiments 2 and 3, where the chosen method's *ft-data* performance is lower than that of the full-model method - despite a higher *OOTD* score. We observe in both of these cases that the *ft-data* is objectively and quantitatively more difficult than the *pt-data* (which may also explain (2) above). To quantify their difficulties, we use the autoencoder reconstruction error [19]. We summarize that our proposed method results in the near-optimal *OOTD* performance in **4/4** of the cases. Based on this result, we suggest that our use of an autoencoder might be an effective general approach to address *a priori* the trade-off between *ft-data* and *OOTD* performance.

2.3.1 Encoder vs. Decoder *OOTD* Paired-Difference Statistical Significance Testing

To ensure that similar feature representational power is represented across a pair in our statistical hypothesis tests, we paired matching numbers of blocks, e.g. "encoder 1" with "decoder 4," "encoder 2" (i.e. encoder blocks 1-2) with "decoder 3" (i.e. decoder blocks 4-3), and so on. To ensure equal sample sizes, we ignore encoder block 5, the bottleneck block of the U-Net architecture. $H_0 : \mu_{encoder} = \mu_{decoder}$. $H_1 : \mu_{encoder} \geq \mu_{decoder}$. To evaluate for statistical significance, we use the Wilcoxon Test to compare the differences in the *OOTD* scores between the matching contiguous-block pairs for each experiment. We use a significance level of $\alpha = 0.05$. We calculated the following p-values for experiments 1, 2, 3, and 4, respectively: .1182, .0010, .0010, .0011. Fine-tuning the encoder subsequences resulted in a greater out-of-domain generalization performance on all experiments except for experiment 1. We note that this may be due to the greater number of classes in the *pt-data* enabling the decoder to learn more generalized feature representations, although more experimentation will be needed.

2.3.2 Contiguous Encoder/Decoder Blocks vs. *OOTD* Scores Linear Relationship Statistical Significance Testing

To evaluate the statistical significance of the linear relationship between an increasing number of encoder blocks and the *OOTD* scores, we use the coefficient p-value after fitting an ordinary least squares (OLS) regression line to each of the experiments' samples. For example, for experiment 1, we would first fit an OLS line to the averaged *OOTD* scores in Table 2.3 and then test using the coefficient p-value. We found statistically significant linear relationships across all experiments except for the encoders of experiment 4, which had a relatively constant relationship. Experiment 2 on the decoder side was a statistically significant negative linear relationship, which is discussed in Section **??**. Tables 2.1 and 2.2 show our results. $H_0 : \beta_1 = 0$. $H_1 : \beta_1 \neq 0$.

2.3.3 Autoencoder Reconstruction Experiment Details

It was noted in [19] that statistically similar data produced lower autoencoder reconstruction errors; data with additional noise/outliers often resulted in higher errors. Using that observation, we train a convolutional autoencoder model on each of the datasets separately and note down the average training loss to quantify the difficulty of each of the three training datasets, *h50*, *ph1-12*, and *ph1-22*. Our convolutional autoencoder consists of 3 convolution-max-pool blocks on the encoder side and 3 convolution-upsampling blocks on the decoder side. Each convolution layer had 3x3-dimension kernels followed by ReLU activation. The final layer consisted of 1 channel, for each of the ultrasound images. The convolutional autoencoder was trained for 5 epochs with a batch size of 16, learning rate of .001, and cross entropy loss. We trained the autoencoder on each dataset 5 times and noted down the reconstruction errors as follows: *h50* (11.379 ± .002), *ph1-12* (12.347 ± .001), and *ph1-22* (12.782 ± .001). We further performed Wilcoxon statistical significance tests for the following cases: (1) $H_0 : \mu_{h50} = \mu_{ph1-12}$. $H_1 : \mu_{h50} \leq \mu_{ph1-12}$. (2) $H_0 : \mu_{ph1-12} = \mu_{ph1-22}$. $H_1 : \mu_{ph1-12} \leq \mu_{ph1-22}$ and calculated the p-values: .0253 and .0258, respectively. This follows along with our observations noted in Section **??**; because the *ft-data* was more challenging, the model may have needed additional blocks for a better latent representation. Related patent pending.

Table 2.1: Encoder-side vs. *OOTD* Scores Linearity Statistical Significance Results. Statistical significance at α = 0.05 bolded.

No.	P-Value	95% Confidence Interval
1	**.008**	(.001, .007)
2	**.001**	(.009, .015)
3	**.017**	(.008, .055)
4	.158	(-.002, .013)

Table 2.2: Decoder-side vs. *OOTD* Linearity Results. Note that experiment 2 has a negative linear relationship.

No.	P-Value	95% Confidence Interval
1	**.001**	(.050, .092)
2	**.011**	(-.066, -.013)
3	**.001**	(.020, .022)
4	**.007**	(.007, .031)

Table 2.3: Transfer Learning Experimental Results using Dice Coefficient metric (averaged over 3 runs) comparing across changes in anatomy and imaging settings. Encoder 1 (e-1) means that only the first block in the encoder is used for fine-tuning, encoder 2 (e-2) means 2 blocks in the encoder are used, decoder 4 (d-4) refers to only the first block on the decoder side, and so on. *OOTD* is the arithmetic average of the Dice metric on the unseen datasets.

No.	Model	*pt-data*	*ft-data*	Method	*h50*	*ph1-12*	*ph1-22*	*ph2-12*	*p12*	*OOTD*
1	1	ph1-12	h50	e-1	.499 ± .0003	—	.578 ± .0010	.610 ± .0030	.531 ± .0023	.583 ± .0206
1	2	ph1-12	h50	e-2	.848 ± .0013	—	.614 ± .0001	.592 ± .0113	.596 ± .0001	.600 ± .0116
1	3	ph1-12	h50	e-3	.814 ± .0010	—	.615 ± .0001	.577 ± .0002	.595 ± .0001	.596 ± .0150
1	4	ph1-12	h50	e-4	.913 ± .0164	—	.614 ± .0001	.596 ± .0051	.597 ± .0009	.602 ± .0086
1	5	ph1-12	h50	e-5	.906 ± .0001	—	.614 ± .0001	.582 ± .0001	.609 ± .0005	.602 ± .0128
1	6	ph1-12	h50	d-4	.520 ± .0307	—	.383 ± .0333	.626 ± .0083	.275 ± .0175	.428 ± .1485
1	7	ph1-12	h50	d-3	.759 ± .0016	—	.582 ± .0021	.687 ± .0032	.401 ± .0065	.557 ± .1184
1	8	ph1-12	h50	d-2	.723 ± .0036	—	.599 ± .0031	.690 ± .0020	.392 ± .0063	.560 ± .1247
1	9	ph1-12	h50	d-1	.959 ± .0009	—	.597 ± .0014	.719 ± .0014	.651 ± .0016	.656 ± .0498
1	10	ph1-12	h50	Full	.957 ± .0010	—	.546 ± .0001	.657 ± .0001	.632 ± .0003	.612 ± .0383
2	11	h50	ph1-12	e-1	—	.589 ± .0033	.536 ± .0001	.625 ± .0010	.596 ± .0001	.585 ± .0372
2	12	h50	ph1-12	e-2	—	.491 ± .0003	.545 ± .0001	.577 ± .0002	.623 ± .0001	.582 ± .0319
2	13	h50	ph1-12	e-3	—	.608 ± .0001	.552 ± .0016	.653 ± .0005	.600 ± .0004	.602 ± .0408
2	14	h50	ph1-12	e-4	—	.629 ± 0001	.604 ± .0002	.667 ± .0001	.595 ± .0002	.622 ± .0317
2	15	h50	ph1-12	e-5	—	.656 ± .0006	.605 ± .0034	.673 ± .0001	.596 ± .0001	.625 ± .0346
2	16	h50	ph1-12	d-4	—	.483 ± .0003	.497 ± .0001	.564 ± .0002	.629 ± .0001	.563 ± .0540
2	17	h50	ph1-12	d-3	—	.507 ± .0001	.494 ± .0002	.571 ± .0015	.611 ± .0002	.559 ± .0484
2	18	h50	ph1-12	d-2	—	.508 ± .0003	.491 ± .0014	.557 ± .0005	.601 ± .0001	.550 ± .0452
2	19	h50	ph1-12	d-1	—	.845 ± .0113	.482 ± .0004	.554 ± .0024	.267 ± .0003	.435 ± .1217
2	20	h50	ph1-12	Full	—	.919 ± .0002	.366 ± .0017	.417 ± .0041	.201 ± .0011	.328 ± .0919

Table 2.4: Transfer Learning Experimental Results using Dice Coefficient metric (averaged over 3 runs) comparing across imaging settings. Encoder 1 (e-1) means that only the first block in the encoder is used for fine-tuning, encoder 2 (e-2) means 2 blocks in the encoder are used, decoder 4 (d-4) refers to only the first block on the decoder side, and so on. *OOTD* is the arithmetic average of the Dice metric on the unseen datasets.

No.	Model	*pt-data*	*ft-data*	Method	*h50*	*ph1-12*	*ph1-22*	*ph2-12*	*p12*	*OOTD*
3	21	ph1-12	ph1-22	e-1	.456 ± .0032	—	.561 ± .561	.627 ± .0004	.325 ± .0016	.469 ± .1239
3	22	ph1-12	ph1-22	e-2	.769 ± .0013	—	.612 ± .0015	.662 ± .0019	.419 ± .0016	.617 ± .1467
3	23	ph1-12	ph1-22	e-3	.794 ± .0003	—	.773 ± .0009	.647 ± .0005	.410 ± .0050	.617 ± .1582
3	24	ph1-12	ph1-22	e-4	.700 ± .0025	—	.838 ± .0032	.594 ± .0168	.595 ± .0040	.630 ± .0511
3	25	ph1-12	ph1-22	e-5	.729 ± .0042	—	.852 ± .0040	.601 ± .0059	.531 ± .0064	.620 ± .0820
3	26	ph1-12	ph1-22	d-4	.179 ± .0007	—	.753 ± .0045	.375 ± .0004	.200 ± .0008	.251 ± .0878
3	27	ph1-12	ph1-22	d-3	.201 ± .0001	—	.840 ± .0029	.403 ± .0006	.218 ± .0001	.274 ± .0916
3	28	ph1-12	ph1-22	d-2	.218 ± .0004	—	.849 ± .0058	.419 ± .0038	.238 ± .0002	.292 ± .0904
3	29	ph1-12	ph1-22	d-1	.256 ± .0015	—	.851 ± .0029	.464 ± .0006	.221 ± .0002	.314 ± .1075
3	30	ph1-12	ph1-22	Full	.481 ± .0151	—	.923 ± .0001	.471 ± .0025	.227 ± .0009	.393 ± .1175
4	31	ph1-22	ph1-12	e-1	.597 ± .0004	.684 ± .0002	—	.539 ± .0018	.375 ± .0023	.504 ± .0943
4	32	ph1-22	ph1-12	e-2	.664 ± .0005	.708 ± .0006	—	.592 ± .0515	.344 ± .0001	.533 ± .1541
4	33	ph1-22	ph1-12	e-3	.650 ± .0042	.823 ± .0002	—	.595 ± .0026	.315 ± .0015	.520 ± .1466
4	34	ph1-22	ph1-12	e-4	.545 ± .0009	.936 ± .0034	—	.642 ± .0013	.326 ± .0024	.505 ± .1605
4	35	ph1-22	ph1-12	e-5	.555 ± .0071	.969 ± .0001	—	.732 ± .0019	.338 ± .0047	.541 ± .1702
4	36	ph1-22	ph1-12	d-4	.193 ± .0001	.740 ± .0038	—	.579 ± .0002	.234 ± .0002	.335 ± .1729
4	37	ph1-22	ph1-12	d-3	.233 ± .0005	.816 ± .0019	—	.498 ± .0017	.204 ± .0005	.312 ± .1320
4	38	ph1-22	ph1-12	d-2	.269 ± .0005	.826 ± .0002	—	.604 ± .0011	.216 ± .0001	.363 ± .1716
4	39	ph1-22	ph1-12	d-1	.329 ± .0002	.965 ± .0001	—	.543 ± .0022	.241 ± .0011	.371 ± .1275
4	40	ph1-22	ph1-12	Full	.370 ± .0024	.958 ± .0001	—	.634 ± .0002	.203 ± .0009	.403 ± .1776

Table 2.5: Proposed Method for Enhanced Generalization using *OOTD* Scores

No.	Ours (Method)	Full
1	**.656 ± .0498** (d-1)	.612 ± .0383
2	**.625 ± .0346** (e-5)	.328 ± .0919
3	**.620 ± .0820** (e-5)	.393 ± .1175
4	**.541 ± .1702** (e-5)	.403 ± .1776

Chapter 3

Stochastic Temporal Data Augmentation for Adaptation to Out-of-Distribution Temporal Features

Chapter 3 is adapted from the manuscript:
Edward Chen, Tejas Sudharshan Mathai, Howie Choset, and John Galeotti, "Stochastic Temporal Data Augmentation for Adaption to Out-of-Distribution Temporal Features"
Edward Chen's contributions to the manuscript include: conducting initial literature review, developing the algorithm and experiments, conducting the experiments, analyzing the data, writing the manuscripts and responding to reviewers with revisions. Tejas Sudharshan Mathai assisted with providing experimental suggestions. Howie Choset and John Galeotti are the supervising faculty advisors.

3.1 Introduction

Ultrasound (US) is a portable, cost-effective, and radiation-free [20] imaging modality, in contrast to other modalities, such as CT and MRI. Emergency treatment for traumatic injuries often involves catheter placement, e.g. for dialysis, extracorporeal membrane oxygenation (ECMO), or resuscitative endovascular balloon occlusion of the Aorta (REBOA) for hemorrhage control. The latter two rely on real-time US imaging to identify anatomical landmarks for femoral vascular access. [21].

Prior work in this space has focused on localizing and segmenting femoral region landmarks [22–24]. However, they are often unable to generalize to out-of-training-distribution data, thereby significantly increasing the risk of complications in emergency field scenarios [25]. In an attempt to address the generalization issue, prior approaches [26, 27] have introduced various image augmentation methods to be uniformly and spatially applied across the US images, with the goal to train the model across additional variances in the data. Although such methods work well with consistent changes across the spatial features in US images, they fail to adequately account for the inherent temporal nature of US.

In emergency scenarios, responders may scan patients with rapidly pulsating vessels in an erratic fashion. It is critical that models generalize well across such unpredictable sequences. But in practice, with the challenges of obtaining labelled US data, especially to different scanning rates and pulsating conditions, it is often the case that a dataset contains very few temporal shifts. Several temporal data augmentation methods do exist in literature [28–31]. Window warping [29], in particular, proposes to address the temporal shifts in data by dropping a constant number of data points within each data sequence, thereby expanding to wider temporal horizons. However, window warping and other methods fail to address rapid and/or unpredictable changing shifts in US imaging while searching for anatomical landmarks in emergency scenarios.

Thus, a key motivation of this work is to address the stochastic nature in US scanning during emergency field medicine. In this study, we detail several novel temporal data augmentation techniques aimed at addressing such scenarios. Our temporal augmentation strategies address the changes in US scanning rates by *stochastically* dropping frames within each sequence, assuming both independently and dependently related frames. A major assumption, though, of such augmentation strategies is that the original training data consists of long temporal horizons. We further attempt to address cases where that assumption may not entirely hold true by using non-uniform spatial augmentations across the temporal sequence, thus synthetically instilling motion into the data. In contrast to current data augmentation procedures [29], we show that our approaches generalize better to rapidly-changing US sequences.

Dataset Names. We note that, in this study, *phantom1-multi12* is referred to as *standard-POCUS*, *phantom1-multi22* is referred to as *higher-depth*, and *phantom2-multi12* is referred to as *right-side*.

Contributions. 1) To the best of our knowledge, we are the first to propose three temporal augmentation strategies to address the stochasticity in US scanning with applications in emergency medical operations. 2) We further address the case where there is a short time horizon inherent within the training data. 3) We demonstrate the consistency of all of our approaches through extensive validation on data acquired from different US scanners and settings.

3.2 Methods

When acquiring sequential frames for analysis, it is customary to move the probe with consistent speed and direction. However, especially in emergency scenarios, the probe is moved quickly and irregularly when searching for the right anatomy. To better train for this type of movement, we synthetically augment the training data across spatial-temporal differences in anatomy between ultrasound imaging frames using the 3 methods described below. Ultrasound frame numbering begins at 0 and ends at t. The original and generated sequences are denoted as $orig$ and gen, respectively.

Strategy 1 - Stochastic Frame-Independent Augmentation

This temporal augmentation method assigns a probability value, drawn from a Uniform distribution, to each ultrasound frame. The threshold value on the probabilities, p_t, is set as a manually tuned parameter and is used to determine the frames needed to be kept for the generated sequence, up to the input sequence length.

$$gen = \{orig[i] \ \forall \ i \text{ if } p(orig[i]) > p_t\} \tag{3.1}$$

Strategy 2 - Stochastic Frame-Dependent Augmentation

At each iteration i starting at frame index k, generate a random number f_{i+1} between 1 and f_{range}. The next iteration $i + 1$ starts at index $k + f_{i+1}$ and selects frame $k + f_{i+1}$ to be added into the generated sequence. Once $k + f_{i+1}$ exceeds the number of original frames, the algorithm completes. The algorithm is depicted in Algorithm 1.

Algorithm 1 Stochastic Frame-Dependent Augmentation

```
$t \leftarrow$ total number of frames
$f_{range} \leftarrow$ size of frame range to select from
$orig \leftarrow$ original sequence of frames
$gen \leftarrow []$
$i \leftarrow -1$
while $i < t$ do
    $frame_{pick} \leftarrow$ random(1,$f_{range} + 1$)
    $i = i + frame_{pick}$
    if $i > t$ then
        $break$
    else
        $gen \leftarrow append(orig[i])$
    end if
end while
```

Strategy 3 - Spatially-Shifted Temporal Augmentation

An underlying assumption that the previous two strategies, along with others [29], make is that the original ultrasound sequence consists of noticeable temporal movement. However, in some real-world cases, the movement of the ultrasound transducer or vessel pulsations may be hardly noticable, nearly reducing down to still images. Rather than again going through an extensive data collection and labelling process, we outline our proposed algorithm below.

For each image sequence of length τ, we randomly generate the magnitude of the temporal shift for the sequence Δ to be within the values σ_{lower} and σ_{upper}, which we designate as hyperparameters. We then split [0, Δ] into τ equally-sized time blocks and generate a random integer between each of those blocks. This results in τ pseudo-randomly generated integers, δ, which represent the spatial shift magnitude of each image in the sequence. We additionally randomize each sequence of δ to be either entirely positive or negative. These integers are currently used for three different variations of this strategy: 1) shift the height of each image i in the original sequence by δ_i, 2) shift the width of each image i by δ_i, and 3) randomly shift 1) or 2) for each sequence.

Algorithm 2 Spatially-Shifted Temporal Augmentation

```
$t \leftarrow$ total number of frames
$orig \leftarrow$ original sequence of frames
$gen \leftarrow []$
$\Delta \leftarrow$ *random*($\sigma_{lower}$, $\sigma_{upper}$)
$\beta \leftarrow [0, \frac{\Delta}{\tau}, \frac{2\Delta}{\tau}, ..., \Delta]$
$\alpha \leftarrow$ *random([-1, 1])*
$i \leftarrow 1$
**while** $i < t$ **do**
    $\delta_i \leftarrow$ *random*($\beta_{i-1}$, $\beta_i$)
    $gen_i \leftarrow shift(orig_i, \alpha * \delta_i)$
**end while**
```

3.3 Experiments

We attempted to model a subset of real-world distribution of ultrasound imaging data which is relevant to us, especially in emergency medical scenarios. All data was acquired from a physically realistic CAE Blue Phantom femoral vascular access lower torso ultrasound training model (BPF1500-HP). For imaging the phantom, we used a Fukuda Denshi portable (i.e. Point of Care Ultrasound, *POCUS*) scanner with a 5-12 MHz transducer imaging a maximum depth of 5cm and 10cm respectively. **Phantom-Based Categories**: Four categories of imaged sequences were acquired from the phantom: (1) *standard-POCUS*, (2) *higher-depth*, (3) *sudden-changes*, and (4) *right-side*. **Scanned Anatomies:** Only 1 of the phantom-based categories was collected from the right side of the phantom, which contained different anatomy (muscles, liver etc.) as well as other artifacts unknown to the model. We refer to this as the *right-side* dataset, as opposed to all the others which are from the left side. **Sporadic Motion:** Extensive probe motions were only used when acquiring sequences in the *sudden-changes* category, such as fast, irregular and erratic probe motion, and out-of-plane deformation. **Imaging Settings:** The remaining 2 categories represent different imaging settings: Sequences in *standard-POCUS* were acquired at a depth of 5cm in contrast to the sequences in *higher-depth*, which were acquired at a depth of 10cm (also POCUS). **Data Quantity:** The *standard-POCUS* category consisted of 12 sequences, each containing 50 frames totalling 600 frames. The *higher-depth*, *sudden-changes*, and *right-side* categories consisted of 3 sequences each, with each sequence containing 50 frames totalling 150 frames. All classes (arteries, veins, ligaments and nerves) were present in (at least some frames of) every sequence in every category, and each sequence was annotated by a single expert.

Network Architecture. The deep learning model we use is a 3D U-Net [32] model consisting of 4 encoder blocks (including the bottleneck layer as the 4th encoder) and 3 decoder blocks. Each encoder block consists of 2 pairs of 3D convolutional layers followed by batch normalization and ReLu activation, with a downsampling layer as the last layer of the block. Each decoder block consists of an upsampling layer followed by 2 pairs of convolution layers, batch normalization, and ReLu activation as described in the encoder block. Each convolutional layer consisted of 3x3x3 kernel dimensions. A diagram of the model is shown in Figure 3.2.

Training Details. The model was trained on *standard-POCUS*. Each input sequence of images consisted of 8 ultrasound frames, all resized to 256x256 pixels. Due to memory limitations, we used a batch size of 8. Each original dataset was augmented using simple spatial image transformations: horizontal and vertical flips, gamma adjustment, Gaussian

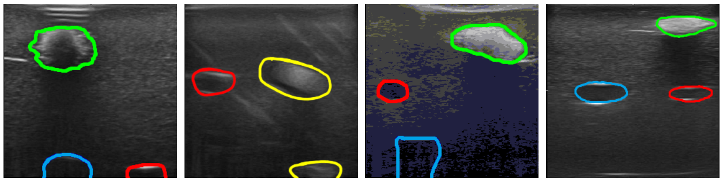


Figure 3.1: Sample images from all of the data categories used in this study. Starting from left: *standard-POCUS*, *sudden-changes*, *right-side*, *higher-depth*. **Color Key**: Red: Artery, Green: Ligament, Blue: Vein, Yellow: Nerve.

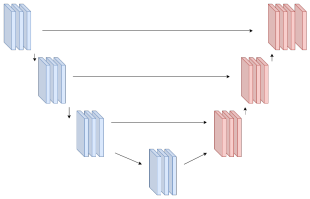


Figure 3.2: 3D U-Net architecture. The encoder blocks (blue) consists of the input layer followed by the 2 pairs of convolutional layer, batch normalization, and ReLu activation. The decoder blocks (red) consists of the same structure. The down and up arrows represent downsampling and upsampling layers.

noise addition, Gaussian blurring, Bilateral blurring, cropping, affine transformations, and shear transformations. The validation set for each experiment was about 10% the size of the original training set. The loss function used was cross entropy loss and model was trained using the Adam optimizer [33]. We set the learning rate to 10^{-4} and each of the networks were trained until the loss did not improve for 6 epochs - using early stopping. The network with the lowest validation loss was then used for evaluation on the testing set.
Baseline Comparisons. We validated our proposed temporal augmentation strategies, *strategy-1*, *strategy-2*, and *strategy-3* against both the traditional spatial augmentation methods, *spatial-only* and another temporal strategy, *window-warp* [29]. We additionally compared against a variation of all of our strategies together; we first randomly selected between temporal strategy 1 and 2, then randomly selected between height and width shift in temporal strategy 3, and we refer to this as *strategy-rand*. The results are shown in Tables 4.1, 3.2, and 3.3. Each of the comparisons were run 2 times.
Ablation Studies. For each of the baselines and our strategies, we performed ablation studies across the single parameter for each of them. For *window-warp*, we only kept every k frame in the sequence where k ∈ [2, 5]. For *strategy-1*, we varied p_t ∈ [0.1, 0.9]. For *strategy-2*, we varied f_{range} ∈ [2, 5]. For *strategy-3*, we kept σ_{lower} as 1 and varied σ_{upper} ∈ [30, 120]. We additionally varied across each of those combinations for *strategy-rand*.
Metrics. We evaluated our experiments using the following metrics: 1) region similarity and 2) contour accuracy. Region similarity measures the similarity of the inner spatial region for each contour between the label and predicted image, whereas contour accuracy measures directly the boundaries of each contour. We calculate the methods as described in [34], where region similarity is the intersection-over-union and contour accuracy is the F-measure over the precision and recall of the contour points between the predicted image and ground-truth.

Table 3.1: Region similarity comparing the best across all ablations, on a dataset in the same domain, *standard-POCUS*, and a dataset with temporal shifts, *sudden-changes*. Bolded values represent our methods which surpassed the results from the baselines.

Approach	*standard-POCUS*	*sudden-changes*
spatial-only	.528 ± .003	.545 ± .005
window-warp	.563 ± .002	.534 ± .009
strategy-1	**.564 ± .003**	**.550 ± .007**
strategy-2	**.639 ± .011**	.534 ± .010
strategy-3	**.584 ± .008**	**.592 ± .009**
strategy-rand	**.638 ± .010**	**.553 ± .012**

Table 3.2: Contour accuracy comparing the best across all ablations, on a dataset in the same domain, *standard-POCUS*, and a dataset with temporal shifts, *sudden-changes*. Bolded values represent our methods which surpassed the results from the baselines.

Approach	*standard-POCUS*	*sudden-changes*
spatial-only	.320 ± .001	.377 ± .002
window-warp	.388 ± .002	.357 ± .003
strategy-1	.362 ± .004	**.390 ± .002**
strategy-2	**.457 ± .007**	**.381 ± .004**
strategy-3	**.412 ± .005**	**.440 ± .006**
strategy-rand	**.432 ± .006**	**.379 ± .004**

Table 3.3: Region similarity comparing the best across all ablations, on datasets outside of training domain. Bolded values represent our methods which surpassed the results from the baselines.

Approach	*right-side*	*higher-depth*
spatial-only	.445 ± .012	.352 ± .011
window-warp	.428 ± .013	.366 ± .012
strategy-1	**.523 ± .010**	.363 ± .009
strategy-2	**.457 ± .009**	**.436 ± .014**
strategy-3	**.579 ± .018**	**.443 ± .015**
strategy-rand	**.484 ± .012**	**.379 ± .010**

3.4 Results and Discussion

From Tables 4.1, 3.2, and 3.3, we noted that our methods outperformed traditional *spatial-only* augmentation method [35] and the *window-warp* temporal augmentation method [29], in most cases. Since all of the experiments were trained on *standard-POCUS* dataset, it was critical that our approach did not hurt the performance on data within that same domain. From Table 4.1, all of our methods surpass both baseline methods, which supports that. In our experiments evaluating against instances with more unpredictable scanning sequences (right column of Table 4.1), all of our methods surpassed the baselines except for *strategy-2*, which performed similarly to the *window-warp* approach. We attribute this to the potential influence of a dependence relation between frames generated by *strategy-2*, due to the subsequent selected frames being dependent on the prior one. In this case, such a relationship seemed to not have been able to capture the more unpredictable nature in *sudden-changes*. A similar pattern is reflected in Table 3.2.

When measured using the contour accuracy metric in Table 3.2, our methods performed significantly better on *standard-POCUS*. Although not the direct intended purpose, we attribute this finding to the additional stochasticity in the training data helping the model learn more generalized features pertaining to its training domain. Similar results on the relationship between stochasticity and generalizability can be seen in Table 4.1 and in existing literature [36–38]. We further note that all of our methods surpassed all baselines on *sudden-changes* across both metrics we used.

To check the consistency of our methods' results across out-of-training domain datasets of different anatomical variations and imaging settings, we evaluated them against *right-side* and *higher-depth* in Table 3.3. We note that our methods outperformed traditional spatial augmentations in all of our experiments, by a large margin. Furthermore, all but one of our methods consistently outperformed *window-warp* as well. We attribute the slightly lower results from *strategy-1* to the lack of frame dependence reflected in the method. The more consistent and stable scanning methodology used when acquiring *standard-POCUS* and *higher-depth* presents additional structure in the collected ultrasound frames which could not have been instilled by *strategy-1* due to its inherent randomness. We note that *right-side*, although collected with a similar structured ultrasound scanning process, contains more variety in anatomy and hence may decrease the relationship between frames. A similar result is also reflected with *strategy-1* in the left column of Table 3.2.

From our ablation studies, we noted some interesting observations. For *strategy-1*, a p_t value of [0.1, 0.2] was too low, either significantly slowing down training or halting it completely. A p_t value closer to 0.5 produced the highest results. For *strategy-2*, a higher value of f_{range}, resulting in a longer time horizon, produced the highest results in our experiments. For *strategy-3*, σ_{upper} values closer to 60 pixels resulted in better performance. Similar patterns were noted when tweaking the same parameters, while keeping the others frozen, for *strategy-rand*. We would like to note that these numerical values may be specific to the features in our datasets and that more experiments will need to be conducted to confirm that.

3.5 Conclusion and Future Work

To the best of our knowledge, we have presented the first three temporal augmentation strategies targeted for medical emergency scenarios with ultrasound imaging. Our results show that, across all of our presented experiments, our novel stochastic methods outperformed the baselines 21 out of 24 times. In the future, we plan to experiment with additional variants of temporal augmentation strategies in addition to making it adaptive to the model training phase.

Chapter 4

Uncertainty-based Adaptive Data Augmentation for Ultrasound Anatomical Variations

Chapter 4 is adapted from the publication:
Edward Chen, Howie Choset, and John Galeotti, "Uncertainty-based Adaptive Data Augmentation for Ultrasound Anatomical Variations," IEEE International Symposium on Biomedical Imaging (ISBI), 2021 (Oral)
Edward Chen's contributions to the manuscript include: conducting initial literature review, developing the algorithm and experiments, conducting the experiments, analyzing the data, writing the manuscripts and responding to reviewers with revisions. Howie Choset and John Galeotti are the supervising faculty advisors.

4.1 Introduction

In the case of high-tempo, traumatic scenarios on the battlefield, real-time ultrasound (US) imaging serves as an enabler for countless possible robotic interventions. Having the ability to automatically segment anatomical landmarks in the body, such as arteries, veins, ligaments, and veins, for percutaneous procedures remains to be a difficult task when considering the countless domains across body types, potential traumatic injury scenarios, and imaging artifacts. Collecting data spanning all, or many, of the cases is tremendously time-consuming and expensive. *A key motivation of this work is to propose a method for enhancing deep learning models' generalization capabilities by generating synthetic data which is transformed in a manner designed to account for various body types, injury scenarios, and imaging features.*

A great amount of the focus in the medical imaging community has been towards engineering improved deep learning architectures, with the goal of learning improved features [39–41]. On the other hand, the data augmentation is a relatively simple and commonly used method for generating synthetic data to account for invariances in the data [42]. Unfortunately, many data augmentation procedures currently rely on domain expertise and manual tuning to be effective [43–45]. Under the manually designed image augmentations, the generated synthetic data may produce countless simple images which limit the model's ability to learn generalized features [43, 44]. In addition, many of the commonly used medical imaging data augmentation strategies still remains to be basic transformations such as flipping, rotating, shifting, and blurring. Although advanced data augmentation strategies do exist for natural images [43–46], many of them are designed for cases which aren't as relevant for medical images. Our goal is to research a learning-based data augmentation method which can adaptively generate augmented images for learning invariances across various anatomical shapes and imaging artifacts.

In this study, we propose a novel data augmentation technique for ultrasound images. Specifically, the method is designed to adaptively train a semantic segmentation network such that it's able to generalize to different anatomical variations, artifacts, and potentially backgrounds. We first employ a Bayesian temporal-based segmentation network which is able to output both the segmentation and epistemic uncertainty [47–49] maps. The epistemic uncertainty maps, which we use to obtain knowledge of the model's spatial weaknesses, are then passed into an augmentation module. Inspired by [50], the augmentation module then initializes a set of fiducial points across the training image and uses an agent feedforward neural network, with the uncerainty maps as input, to create a final moving state for the fiducial points. Similar to [45], we use a similarity transformation based on the moving least squares transformation method [50]. The constructed images are designed to "spot-augment" the images in such a way that they challenge the models precisely where they already have learned good features for, and display less uncertainty. We train the agent neural network based on how much it's able to challenge the base segmentation network, which is measured by the loss on the augmented images. All of these steps are embedded within a unified real-time training framework, shown in Figure 5.2. Our primary contributions are: (1) incorporating the epistemic uncertainty map outputs from a Bayesian segmentation network for "spot-augmenting" areas where the model is currently strong, (2) an adaptive training pipeline which also incorporates the spatial relationship nuances between ultrasound imaging anatomical landmarks, and (3) experiments illustrating how the method is able to better enable medical segmentation networks to generalize across various vessel shapes and imaging features.

Dataset Names. We note that, in this study, *phantom1-multi12* is referred to as *standard-POCUS* and *phantom1-multi22* is referred to as *higher-depth*.

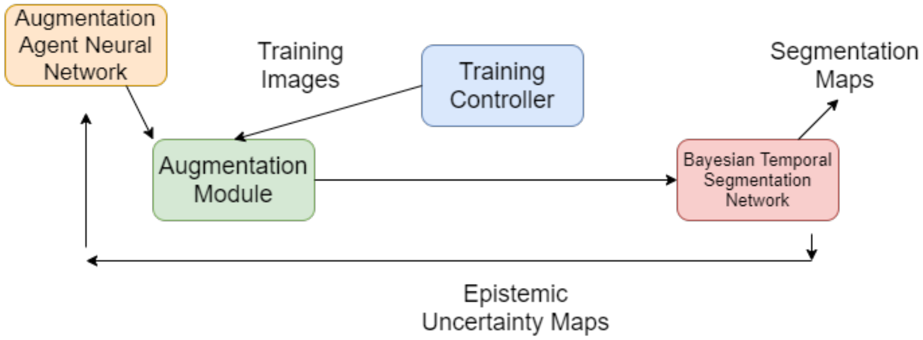

Augmentation Agent Neural Network
Training Images
Training Controller
Segmentation Maps
Augmentation Module
Bayesian Temporal Segmentation Network
Epistemic Uncertainty Maps

Figure 4.1: Flow diagram of the overall uncertainty-based augmentation pipeline. The uncertainty maps from the segmentation network get passed into the agent neural network, which then generates the control points for deforming the image in the augmentation module. Those are then fed back into the training process.

4.2 Methods

4.2.1 Semantic Segmentation Neural Network

Our final task is multi-class, multi-instance segmentation of arteries, veins, ligaments, and nerves in ultrasound images. The deep learning network architecture we use is a dropout-based Bayesian formulation [47, 49] of the 3D U-Net encoder-decoder architecture [32]. The network consists of four convolutional blocks on the encoder side, with matching pairs on the decoder side. Each block consists of an input layer followed by 2 pairs of the following: convolutional layer, batch normalization, and ReLU. Within each block, we empirically determined to place a single dropout layer before the output layer, as opposed to other possible variations [51]. The model outputs two values (represented below), for both the predicted mean, $\hat{\mu}$, the segmentation map, and predicted variance, $\hat{\sigma}^2$, which we use for the epistemic uncertainty map:

$$[\hat{\mu}, \hat{\sigma}^2] = f^{\hat{W}}(x) \tag{4.1}$$

where f is the Bayesian 3D U-Net, in this case, parameterised by model weights $\hat{W}$. The epistemic uncertainty maps are obtained using test-time stochastic forward passes, also referred to as Monte Carlo dropout [49]:

$$\frac{1}{T}\sum_{t=1}^{T}(\hat{\mu}_t - \bar{\mu})^{\bigotimes 2} \tag{4.2}$$

where T is the total number of Monte Carlo samples and $\bar{\mu} = \sum_{t=1}^{T}\frac{\hat{\mu}_t}{T}$. Despite not actively using the logits variance output for computing the aleatoric uncertainty, which is the statistical uncertainty inherent in the data, empirical trials and [49] both illustrated that having the variance output was still necessary. Without the logits variance output, we empirically found that the epistemic uncertainty tended to overcompensate for that fact and obtained poor performance.

4.2.2 Augmentation Module

The augmentation module is responsible for generating the synthetic images for further training. It does so by using a variation of the moving least squares deformation method [50, 52] which first generates a set of control points p around the border of the overall image as well as of the individual anatomical classes. The immediate next step is then to also generate a set of deformed control points q. The points q govern the image deformation using the best affine transformation $l_v(x)$ which minimizes the following [50] :

$$\sum_i w_i |l_v(p_i) - q_i|^2 \tag{4.3}$$

where w_i represents the set of deformation weights, which are dependent on the point of evaluation v [50].

To generate the set of points q, we use a convolutional neural network, which we refer to as the augmentation agent neural network, which takes as input the epistemic uncertainty map and outputs a set of directions to shift the original points p. We output the directions rather than individual shift magnitudes because we found it empirically simpler to train and to avoid potential negative image augmentations. Currently, we only use straight top-down and left-right directional shifts but this can be expanded into additional degrees of freedom as well.

We then apply points p and q to the original training images in the batch to output a new batch with transformed versions of the images. The moving least squares image deformation method is notoriously known to take a long time to compute [52]. To enable it for real-time capabilities in our method, we rearrange the mathematical relationships in such a way that we can pre-compute most of the expensive matrix multiplications prior to the training process. We first represent $l_v(x)$ as an affine transformation with a linear transformation matrix, M, and a translation value, T. According to the proof in [50], we can solve for T by rearranging the relationship as such:

$$T = q_* - p_* M \tag{4.4}$$

where q_* and p_* are the weighted centroids used for the linear moving least squares deformation, represented as such:

$$p_* = \frac{\sum_i w_i p_i}{\sum_i w_i} \tag{4.5}$$

$$q_* = \frac{\sum_i w_i q_i}{\sum_i w_i} \tag{4.6}$$

We re-formulate the above relationships by splitting our initial and final control points, p and q, respectively into a set for the border of the ultrasound image, p_B and q_B, and another set for the anatomical classes within, p_i and q_i. The reason we have a set of dedicated control points for the border is to prevent the sides of the image from folding in, creating holes in the image. We show an example of a proper deformation in Figure 4.2. Furthermore, since the borders of the images stay constant throughout the training process, we can re-arrange the equation for q_* as such:

$$q_* = \frac{\sum_i w_i q_i + w_B q_B}{\sum_i w_i + w_B} \tag{4.7}$$

where p_* would follow a similar structure. We further pre-compute the values of w_i, represented as:

$$w_i = \frac{1}{|p_i - v|^{2\alpha}} \tag{4.8}$$

Additional values for computing the affine transformation represented in [50], which do not depend on each individual image, are also pre-computed. Finally, to customize the speed of the image deformations, we set as hyperparameters the count of the control points p_i and p_B.

Augmentation Agent Neural Network

The augmentation network is a simple, custom 3D convolutional neural network which consists of 5 convolutional blocks. The first 3 convolutional blocks each consist of, sequentially, a 3D convolutional layer, a batch normalization layer, ReLu activation, and then a max pooling layer. The last 2 convolutional blocks do not have the max pooling layer. The final convolutional block outputs to a fully connected layer which then outputs a set of points classifying whether to go up or down for each control point.

To train the augmentation agent neural network, we generate a random set of points, signalling up or down for the deformations. We then compute the moving least squares image deformations [50] for both the agent-generated and randomly-generated points and compute the segmentation loss for both sets. If the agent-generated points resulted in a lower segmentation loss, we assume the randomly-generated points as more difficult and assign those as the label for training this network. If the randomly-generated points resulted in a lower loss, however, we assign the opposite direction of the agent-generated points as the label similar to [45]. The rest of the training process is completed as normal. A diagram detailing the entire pipeline is shown in Figure 5.2.

4.3 Experiments and Results

4.3.1 Data

The data was acquired from a CAE Blue Phantom femoral vascular access lower torso ultrasound training model. To collect the data, we used the Fukuda Denshi portable (i.e. Point of Care Ultrasound, *POCUS*) scanner with a 5-12 MHz transducer imaging a maximum depth of 5cm and 10cm, respectively. The main dataset, which we refer to as *standard-POCUS*, used for training consisted of 12 sequences, each containing 50 frames totalling 600 frames. The other dataset, which we refer to as *higher-depth*, consisted of 3 sequences, with each sequence containing 50 frames totalling 150 frames. It was collected using a higher depth, which deformed the anatomy. All classes (arteries, veins, ligaments and nerves) were present in (at least some frames of) every sequence in every category, and each sequence was annotated by a single expert.

4.3.2 Training Details

Due to the memory restrictions presented by our precomputation details, we only train with a batch size of 1, with 8 images within each sequence. Each of the images are resized to 256x256 pixels. Also a consequence of the memory-hungry characteristic of our method, we do not perform initial significant offline data augmentations to the training set. For training, we used the aforementioned Bayesian temporal segmentation network with a stochastic version of the cross entropy loss, similar to [49]. To compute the uncertainty maps, use 10 Monte Carlo samples [49]. We set the learning rate to 10^{-3} and trained the network until the validation loss did not improve for 5 epochs. Lastly, the validation set consisted of about 10% of the original training dataset.

4.3.3 Experiments

We compare our proposed pipeline against the same Bayesian temporal segmentation network across 2 cases: 1) without any spatial data augmentations, so using the same initial training set as our pipeline (*non-aug*), and 2) using a full set of spatial augmentations (*full-spatial-aug*). The full set of spatial augmentations consisted of horizontal and vertical flips, gamma adjustment, Gaussian noise addition, Gaussian blurring, Bilateral blurring, cropping, affine transformations, and shear transformations. In terms of the control points, we use 128 and 2 points for the border and class, respectively. To limit the up/down directions, we use 20, 40, 80, 40 pixels as the maximum shifts for the arteries, veins, ligaments, and nerves, respectively. All of the experiments are evaluated against *standard-POCUS* and *higher-depth* across 2 trials. The results are shown in Table 4.1.

4.3.4 Metrics

To evaluate our experiments, we computed the following metric - region similarity. Region similarity measures how similar the ground-truth label and predicted images are in terms of their inner contour regions. We calculate region similarity as how it is used in [53], using the intersection-over-union between the ground-truth label and predicted image.

Approach	*standard-POCUS*	*higher-depth*
non-aug	.534 ± .010	.238 ± .004
full-spatial-aug	.564 ± .012	.453 ± .009
Ours	**.571** ± **.019**	.419 ± .013

Table 4.1: Region similarity metric comparing the best-performing methods, across the 2 datasets. The bolded values represent our methods which surpassed the baseline results.

4.4 Discussion

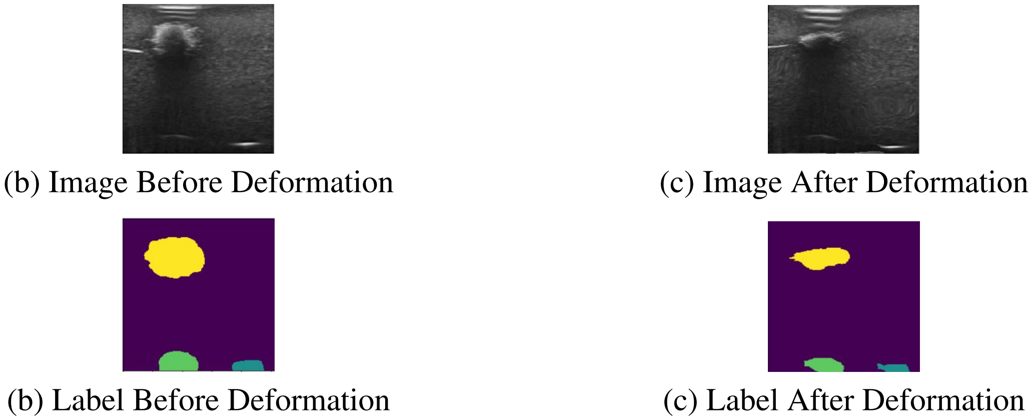
Based on the results presented in Table 4.1, our pipeline is able to enhance the generalization performance of segmentation models due to the diversity of images which it generates. We would like to emphasize that the performance of our method in Table 4.1 only uses the original training set, without any prior spatial augmentations - which is a 28x difference. Even with the lack of augmentation, the method is able to approach the out-of-domain generalization abilities of the fully augmented version, reflected by the score on *higher-depth*. Some samples of the image deformations are shown in Figure 4.2. In terms of computation time, the pipeline is able to completely generate the augmented batch of images in, on average, 10.576 seconds. To completely perform the moving least squares image deformation [50], it takes, on average, 5.201 seconds - which we have to compute twice for both the images and the labels. Those times are all based on the 128 border control points we use, which affects it the most. We noticed that switching it to 256 border points nearly doubles the computation time, hence why we reduced to 128. We reduced the number of class control points to 2, for each present class, for the same reason. The reason for the greater number of border points is because we empirically noticed that reducing the number of border points too much created visible holes near the borders and made the images look unrealistic.

The biggest limitation for this pipeline right now is the aforementioned computational time and memory requirements. In order to pre-compute all of the matrices mentioned, it takes roughly 260 MB of hard drive space for each image, hence our memory restrictions during training it (not being able to use the full spatial augmentations). As with the computation time, this memory requirement also decreases in proportion to the number of border points.

Another potential limitation is that this method may miss image deformations in areas of the image where the segmentation ground-truth label does not cover. As a result, there may be cases where traditional, uniform, spatial augmentations will maintain better generalization capabilities.

4.5 Conclusion and Future Work

To the best of our knowledge, we have presented the first online, adaptive data augmentation pipeline for adapting to different anatomical variations with ultrasound imaging. According to our results, the pipeline is able to enhance the generalizability of the segmentation network, but with a cost due to the memory and computational time. In the future, we plan to improve upon this by expanding the degrees of freedom with which the pipeline is able to modify the features in the ultrasound images. We also plan to improve the computational memory and time requirements of this overall pipeline, in order to make it more feasible for consistent training.



(b) Image Before Deformation

(c) Image After Deformation

(b) Label Before Deformation

(c) Label After Deformation

Figure 4.2: Example ultrasound image undergoing the image deformation. The top images visualize the original and new images, whereas the bottom images visualize the same images' labels.

Chapter 5

Multi-Class Bayesian Segmentation of Robotically Acquired Ultrasound Enabling 3D Site Selection along Femoral Vessels for Planning Safer Needle Insertion

Chapter 5 is adapted from the manuscript:
Edward Chen, Abhimanyu, Vinit Sarode, Howie Choset, and John Galeotti, "Multi-Class Bayesian Segmentation of Robotically Acquired Ultrasound Enabling 3D Site Selection along Femoral Vessels for Planning Safer Needle Insertion"
Edward Chen's contributions to the manuscript include: conducting initial literature review, developing the segmentation and optimal needle insertion algorithm and experiments, conducting the experiments, analyzing the data, writing the manuscripts and responding to reviewers with revisions. Abhimanyu's contributions include developing the scanning and 3D reconstruction algorithms, conducting the corresponding experiments, analyzing the data, and writing the corresponding sections. Vinit Sarode's contributions include helping to run experiments for all of the components. Howie Choset and John Galeotti are the supervising faculty advisors.

5.1 Introduction

Percutaneous, or needle-puncture, procedures are often used for a wide variety of anatomical targets within the body and are typically associated with performing safe and minimally-invasive surgeries. Common applications include central vascular access for resuscitation, arterial pressure monitoring, emergency dialysis catheter placement as well as rarer, more invasive, endovascular interventions, extracorporeal membrane oxygenation (ECMO), and resuscitative endovascular balloon occlusion (REBOA) [54, 55]. In many of those cases, placement of a needle in the proper location is essential to a positive outcome.

Previous literature on endovascular intervention supported that percutaneous femoral

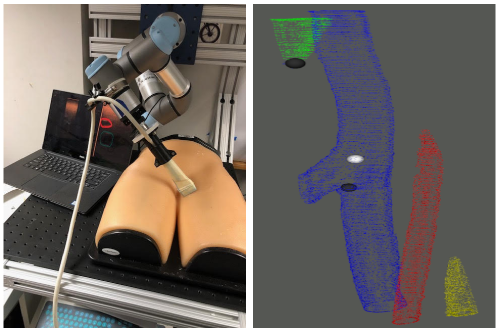


Figure 5.1: Portable robotic system designed for ultrasound-guided needle site selection. The image on the left displays a sample set-up of the pipeline. The image on the right displays the final 3D visualization with the optimal insertion point (white) for the left side of a torso phantom. **Color Key**: Artery - Red, Vein - Blue, Ligament - Green, Nerve - Yellow.

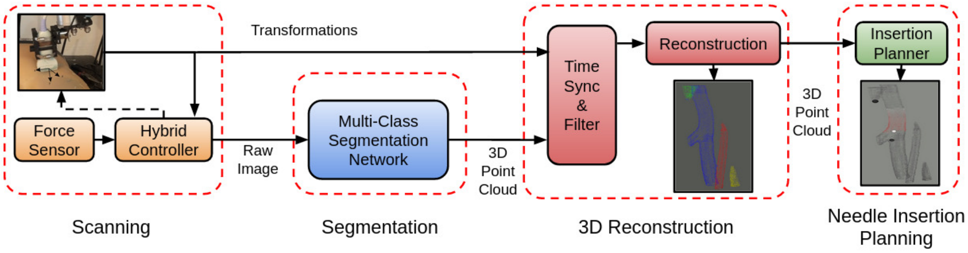


Figure 5.2: Flow diagram of the overall automatic pipeline. The robot system first collects ultrasound images from its scanning process, sends them to the deep learning model for segmentation, retrieves the segmentation coordinates for 3D model generation, and then outputs suggested locations for needle insertion based on safety standards.

arterial access is associated with serious complications [56]. Especially with older patients, complications related to insertion, such as hematomas (2-8%) and pseudoaneurysms (1-2%), are becoming more common with the growing number of procedures done in the femoral area [57]. The risk of such complications are further increased when dealing with high-tempo, stressful situations or less experienced medical clinicians. Furthermore, inaccurate judgement of mental 3D models or ultrasound images often result in multiple punctures, taking more time in critical scenarios. Severe medical issues also arise as a result of needle insertion in other location sites, such as transradial artery and liver access [58]. Automated approaches using robotics can reduce these risks significantly [59].

Portability is key in emergency medical scenarios. Computer vision coupled with the ultrasound imaging modality plays a critical role in the flexibility of medical robotics. Ultrasound is small, low-cost, and field-portable, unlike other imaging techniques such as magnetic resonance imaging (MRI), computed tomography (CT), or X-ray. As a result, the entire robot has the potential of being easily transported to different locations for serving emergency medical purposes.

In this study, we present a fully automatic pipeline for robotic control for vascular needle insertion planning in the femoral region as a major step towards real-world deployment during medical emergencies, outlined in Figure 5.2. The femoral region is used as it allows for rapid administration of medications, critical for emergency situations [60]. A sample set-up and result is shown in Figure 5.1. The robot uses a Bayesian deep learning-based multi-class 3D CNN segmentation network for building a 3D visualization of the tissue in the femoral region, which our needle insertion planning algorithm then uses to determine the safest location for insertion such that the risk of complications will be lowered. For further elucidation of the algorithm results, we also generate 3D heatmap visualizations depicting the needle insertion safety levels, along with uncertainty-based pruning of noisy segmentations. The robot is able to collect ultrasound images on both smooth and curved surfaces while being able to segment arteries, veins, ligaments, and nerves simultaneously. This entire pipeline is able to be performed with zero human intervention, decreasing the expertise required to safely perform a variety of life-saving transcutaneous interventions [61]. As a result, we introduce the following contributions:

1. a novel algorithm for standardizing an optimally safe location for vascular femoral-region needle insertion by using a 3D visualization generated from deep learning-based multi-class segmentations
2. an automatic robotic pipeline capable of scanning curved surfaces with ultrasound, performing multi-class imaging, 3D anatomic visualization, and the above insertion planning algorithm all within a Bayesian framework

The rest of the paper is structured as such: the following section discusses related work. Section III describes each of the individual components in depth: robotic scanning, multi-class segmentation, 3D visualization, and needle insertion planning. Section IV provides results we obtained over repeated trials with multiple different test subjects across various imaging settings and ultrasound scanners. We then conclude with an overview of our methods along with potential avenues for future work.

Dataset Names. We note that, in this study, *phantom1-multi12* is referred to as *torso-left* and *phantom2-multi12* is referred to as *torso-right*.

5.2 Related work

There currently is a wide variety of existing literature which discusses robotic systems for needle insertion tasks [59, 62–67]. We can broadly categorize the approaches by their choice of imaging modality and anatomical landmark localization method. Specifically for the femoral region, most of the existing work targets non-ultrasound based imaging modalities, such as flouroscopy and X-ray [62, 63], both of which expose the patient to ionizing radiation. Ultrasound, which lacks ionizing radiation, is safe for continuous imaging and is also more portable, making it a better choice for our intended use case of emergency scenarios. Despite its lightweight benefits, ultrasound does have the disadvantage of containing more noise in the images compared to those from other modalities. We deal with such characteristics by training a deep learning model based on the 3D U-Net [68, 69], within a Bayesian framework [70, 71], from various augmented images - helping it to automatically learn discriminative features from the images.

Other existing works for localizing the anatomical landmarks rely on more conventional methods such as brute force searches and radii measurements for vessel segmentation [63–65, 72, 73]. The non-deep learning based approaches used in such works rely on either an offline initial setup for registration of an *a priori* anatomic model to the images or a brute force search for the image processing algorithm [63–65, 72, 73]. [73] deals with segmenting nerves in ultrasound images for regional anesthesia, but only does so for the single class, nerves. [59] is one of the more recent works which employ deep learning-based methods for vessel segmentation in a semi-automatic robotic pipeline using dual imaging modalities, near-infrared and ultrasound, for vascular access in the arm region. Our proposed method for determining regional anatomical landmarks for needle insertion uses a Bayesian deep learning-based ultrasound vessel segmentation network that also segments arteries, veins, and ligaments, with no human intervention, and is also able to maintain some generalizability across ultrasound settings. In addition, our proposed method combines the multi-class Bayesian segmentation method in a novel way with our optimal needle insertion planning algorithm.

Some existing approaches for determining the optimally safe insertion location for needles use a geometrical model [65]. Other algorithms don't take into account the location of the inguinal ligament or require manual user input to obtain the anatomical landmarks [59, 64, 66]. In the prior works using a geometric model, the optimal insertion site is determined directly based on a single image rather than with a 3D visualization or sequence of frames across the insertion area. For the femoral region, it is critical to not just use a single image but to have some anatomical model of the area such that the physical relationship of the global anatomical landmarks can be considered [67]. Considering the location of both the inguinal ligament and vessel bifurcation is important for preventing additional complications, such as increased risk of retroperitoneal hematoma or hemorrhage [74, 75].

The success of the aforementioned approaches in needle insertion lend themselves to the advantages of automating the needle insertion process. However, these prior methods have limitations resulting from their various imaging modalities, classical approaches for segmentation, and global anatomical landmark choices. As such, these existing robots have deficiencies that preclude their portable use for real-world emergencies. Our proposed robotic pipeline overcomes some of the deficiencies by essentially learning to segment all relevant anatomical landmarks within the femoral region in a way that could be more easily transferred to other imaging settings or human anatomical variations. To the best of our knowledge, our proposed method is also the first work on a fully automatic robotic system from the scanning phase up to determining the optimal needle insertion site.

5.3 Methods

5.3.1 Experimental Setup

We use the Universal Robot UR3e model for ultrasound scanning. The ultrasound scanning is evaluated with a Fukuda Denshi portable point-of-care ultrasound scanner (POCUS), using a 5-12 MHz 2D transducer. The experiments and data were gathered from a CAE Blue Phantom anthropomorphic gel model, *blue-gel*, both the left and right sides of a CAE Blue Phantom lower torso ultrasound training model BPF1500-HP (which each contain different anatomical variations), *torso-left* and *torso-right*, and a live pig, *live-pig*. The IACUC-approved experiment on the live pig was done in a controlled lab setting under the supervision of clinicians. The deep learning pipeline for multi-class segmentation was built using TensorFlow [76] and Python. Our optimal needle insertion planning algorithm was implemented in Python, and Robot Operating System (ROS) [77] was used to combine all of the components together. The current robotic setup is not yet suitable for real-world emergency situations, requiring substantial future work to address issues of size, sterility, safety/FDA certification, etc.

5.3.2 Robot Controller

Consistent force is necessary for acoustic coupling, patient comfort, and avoidance of excess pressure distorting tissue. Without it, the segmentation performance may be disrupted.

Force Regulation: In our system, the robot is driven by velocity commands. To maintain a constant force, a velocity is applied along the direction of the ultrasound probe [78] (coordinate system shown in Figure 5.2). The applied velocity (which is along the y-direction), v_y, is proportional to the difference between desired force, f_d, and the actual force measurement, f, from the sensor, as shown:

$$v_y = -K_f * (f - f_d) \tag{5.1}$$

where K_f is the force controller gain.

Position Control: We control the position of the robot by manually defining the start and end points of the scanning motion in ROS. The start and end positions are chosen to maximize the anatomical landmark coverage during scanning. The velocity values, $v_{x,z}$, are computed with the following feedback control law [79]:

$$v_{x,z} = -K_{x,z}(p_{x,z} - {p^*}_{x,z}) \tag{5.2}$$

where $K_{x,z}$ is the feedback controller gain for motion in the x- and z-directions, and $p_{x,z}$ and ${p^*}_{x,z}$ are the current and goal locations, respectively, in the xz-plane (which is normal to the ultrasound probe). The end effector velocity is then converted to the target joint velocity, which in turn is sent to the UR3e robot.

5.3.3 Multi-Class Segmentation of Arteries, Veins, Ligaments, and Nerves

The deep learning model we use is a Bayesian formulation of the 3D U-Net encoder-decoder architecture, inspired by [69, 70]. We use a sequence of 8 two-dimensional ultrasound images as input to the model, where the temporal aspect is treated as the third dimension. Due to memory limits, the encoder side of the network consists of four encoder blocks, with each block consisting of 3D convolution, batch normalization, and ReLu layers as described in [69]. The decoder side of the network consists of the encoder-paired decoder blocks [69]. We formulate this 3D U-Net into a Bayesian version [70] by placing a distribution over its weights with a single dropout layer at the output of each encoder and decoder block, which we empirically found to produce the best results. The model then consists of two outputs, one for the predictive mean, $\hat{\mu}$, and another for the predictive variance, $\hat{\sigma}^2$, as represented in [70]:

$$[\hat{\mu}, \hat{\sigma}^2] = f^{\hat{W}}(x) \tag{5.3}$$

where f is the Bayesian 3D U-Net, in this case, parameterised by model weights $\hat{W}$. The model is trained using the stochastic cross entropy loss formulated in [70]. Epistemic uncertainty maps, which represent the model uncertainty, are obtained using test-time stochastic forward passes, also referred to as Monte Carlo dropout [70]:

$$\frac{1}{T}\sum_{t=1}^{T}(\hat{\mu}_t - \bar{\mu})^{\otimes 2} \tag{5.4}$$

where T is the total number of Monte Carlo samples and $\bar{\mu} = \sum_{t=1}^{T} \frac{\hat{\mu}_t}{T}$. Despite not actively using the logits variance output, $\hat{\sigma^2}$, for computing the aleatoric uncertainty, statistical uncertainty inherent in the data, empirical trials and [70] both illustrated that having the variance output was still necessary. Without the logits variance output, we found that the epistemic uncertainty tended to overcompensate for that fact and obtained poor performance.

To further account for variability in ultrasound imaging, data augmentation in the form of rotations, translations, flips (up-down and left-right), zooms (in and out), filtering, and blurring was applied to the data prior to training the model.

For training of the model, we used the Adam optimizer [80], a learning rate value of 0.0001, a sequence length of 8 frames, and a batch size of 8. We use *256x256* for the image dimensions. For obtaining the epistemic uncertainty maps, we use $T = 2$ Monte Carlo samples, due to time constraints. A diagram is shown in Figure 5.3.

5.3.4 3D Visualization of Multi-Class Segmentation

Upon obtaining the segmentation and uncertainty maps from the deep learning model, we generate a 3D point cloud visualization of the anatomical landmarks in the scanned region, with noisy segmentations filtered out. To filter out false-positive segmentation results, we propose the following: (1) calculate the average uncertainty values, υ_i, within

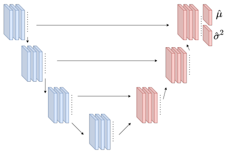


Figure 5.3: Diagram of our Bayesian 3D U-Net model. The blocks represent 3D convolutions + Batch Normalization + ReLu, the dotted lines represent dropout, and the final outputs represent the predictive mean and variance.

every segmentation contour, and (2) filter υ_i by class, c, and calculate uncertainty thresholds, τ_c, with:

$$\tau_c = \hat{\upsilon_c} + \hat{\sigma_c} * \delta \tag{5.5}$$

where $\hat{\upsilon_c}$ and $\hat{\sigma_c}$ are the average and standard deviation, respectively, of υ_i taken for class c, and δ is a manually tuned parameter representing the number of standard deviations away from the mean to filter out. In practice, we found that the PERT statistical distribution [81] provided the best approximation to the uncertainty values υ_i.

To obtain the coordinates for plotting with respect to robot's fixed base frame, ${}^{ro}P_{im/px}$, we apply transformations from the segmented ultrasound image as follows:

$${}^{ro}P_{im/px} = {}^{ro}T_{tr} * {}^{tr}T_{im/mm} * {}^{im/mm}S_{im/px} * p_{im/px} \tag{5.6}$$

where $p_{im/px}$ is the segmented region in the image, ${}^{im/mm}S_{im/px}$ is the scaling factor to convert from pixel to millimeter (mm) units, ${}^{tr}T_{im/mm}$ is the transformation matrix to put the mm units into respect with the ultrasound transducer's frame, and ${}^{ro}T_{tr}$ is the transformation relation applied to place the points from the ultrasound transducer's frame into the robot's fixed base frame. ${}^{tr}T_{im/mm}$ is obtained from the manual calibration procedure described in [82] and ${}^{ro}T_{tr}$ is obtained from the *tf* ROS package.

The color encoding scheme used in the 3D visualization is as follows: **artery** - red, **vein** - blue, **ligament** - green, **nerve** - yellow. Examples of our 3D generated models are shown in Figures 5.1, 5.8, 5.10, and 5.11.

5.3.5 Optimal Needle Insertion Planning Algorithm

We use the femoral artery as the target vessel in this case, but it can be easily extended to the femoral vein or even other tissues. A diagram of the anatomy is displayed in Figure 5.4. The ideal site for femoral arterial puncture is generally accepted to be over the femoral head, below the inguinal ligament, and above the femoral arterial bifurcation [83, 84]. More specifically, recent medical literature has pointed more towards inserting 75% of the way down from the top of the femoral head, hence being closer to the arterial bifurcation (where the inguinal ligament and arterial bifurcation are 33% and 100% from the top of the femoral head, respectively) [85]. Even in cases where doctors are aware of the arterial bifurcation point through ultrasound imaging, they tend to go at least 1 cm cranially [86]. These practices are reflected in our algorithm. To further account for noise in the segmentation

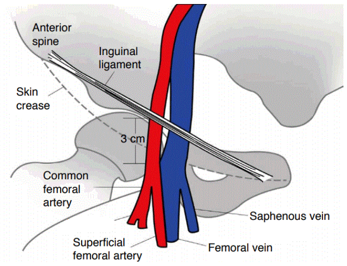


Figure 5.4: Diagram of ideal femoral arterial puncture site [1]. The 3 cm range shown in the figure indicates ideal region for needle insertion into femoral artery. Used Creative Commons Attribution Non-Commercial License.

outputs, we filter out segmentation predictions for class c which have pixel-areas smaller than ϕ_c pixels. In practice, we used 100, 300, and 1000 for ϕ_{artery}, ϕ_{vein}, and $\phi_{ligament}$, respectively, for the algorithm below:

(1) Detect the femoral arterial bifurcation point. We implement this by checking for a gap at least of size g between contours from the artery class. We then assume the location with the smallest gap as the point of bifurcation and refer to this as point α. To account for noise in the segmentation results, we check that at least $\gamma\%$ of the contours caudal to that point also contain a count of at least 2. We empirically determined the values of 3 and 95 for g and γ, respectively.

(2) Detect the caudal end of the inguinal ligament. *If a ligament was detected* in the scan, we determine the closest point on the ligament to point α. *If the ligament was not scanned/detected*, we account for 2 scenarios: (a) the femoral artery follows a straight path, and (b) there exists a gradual curve in the femoral artery, referring to where the artery is crossing under the ligament. For (a), we currently assume the ligament's location to be immediately off the cranial edge of the scan. For (b), we iterate over each arterial contour at index i and calculate the angles between vectors u and v, with opposing endpoints at $i - k$ and $i + k$, respectively, using the following: $\cos\theta = \langle u_k, v_k \rangle / |u_k||v_k|$. An illustration is depicted in Figure 5.11. We then assume the location at index i with the smallest angle θ as the ligament landmark, which we will refer to as point λ. We account for noise similar to the previous step, except by checking for a count of 1.

(3) Determine a safe region in between the two anatomical landmarks. We do this by shifting α and λ towards each other by $\delta_\alpha\%$ and $\delta_\lambda\%$, respectively. We denote these shifted safe boundaries as α_s and λ_s, respectively. We do this (1) to account for noise in the deep learning segmentation outputs and (2) to incorporate common medical practices as mentioned above. We used 15 for the value of δ_α and δ_λ, accounting for the 1 cm minimum distance to the arterial bifurcation given an average common femoral artery (CFA) segment length of $\sim$7 cm [87].

(4) Calculate the percentage of overlap between the femoral vein and artery at all points. We do this by calculating the percentage of overlapping pixels when viewing the contours from a posterior angle. We refer to this as V_o.

(5) Compute scores and determine optimal insertion location. We use the following relations:

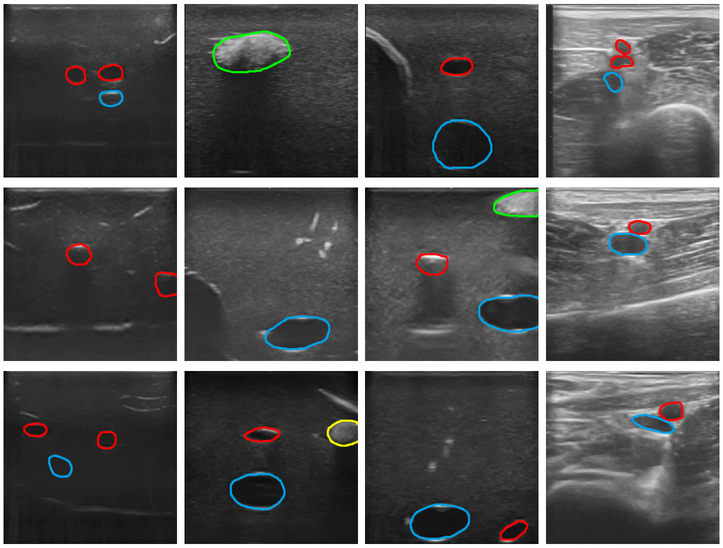


Figure 5.5: Examples of the variety in images for which we validated our methods on, with the labels overlaid on top. From left to right: *blue-gel*, *torso-left*, *torso-right*, *live-pig*. Each row shows the 3 different imaging settings with which we varied for each subject, except for *live-pig*. **Color Key**: Arteries - Red, Veins - Blue, Ligaments - Green, Nerves - Yellow.

$$P_h(\zeta) = \begin{cases} \frac{\sigma_\alpha}{1+\|\zeta-\alpha\|_2} + \frac{\sigma_\lambda}{1+\|\zeta-\lambda\|_2}, & \text{if } \zeta \in [\lambda_s, \alpha_s] \\ \infty, & \text{otherwise} \end{cases}$$

$$T_s = P_h + V_o \tag{5.7}$$

where P_h is the *Proximity Hazard Score* to account for distance from α and λ, ζ is the 3D coordinate for the center of an arterial segmentation contour, σ_α and σ_λ are values for weighing the importance of sufficient distance from α and λ, respectively. We reflect the aforementioned insertion percentiles commonly used in practice with σ_α and σ_λ. Arterial contours not within the safe region or of an area smaller than ϕ_{artery} are assigned a maximum score. The *Total Site Score*, T_s, is then obtained by adding V_o and P_h together, taking into account overflow and underflow. To select the final insertion location, we select the artery corresponding to the lowest value of T_s. If there are multiple of such values, we take the largest-sized artery cross-section.

To enhance clinical viability and explainability, we also generate a second 3D visualization illustrating a heatmap of T_s. We do this using the same steps described in the previous section, except with the following color encoding scheme: non-artery/vein structures and regions with values of ∞ are colored with RGB value (128,128,128), which is grey, and arteries are shaded with RGB values of (255,η,η) where $\eta = min(T_s - min(T_s) * 255, 255)$. As a result, regions with higher values of T_s appear white, whereas lower values appear bright red. The algorithm was implemented using vectorized NumPy Python library methods.

5.3.6 Fully Automatic Pipeline

1. Hybrid force-position controller obtains force/positional feedback from ROS, then used to calculate target joint velocity sent to the UR3e arm.
2. Raw images from the ultrasound scanner are passed to segmentation model via ROS. Segmented regions' coordinates then obtained for 3D visualization.
3. Segmented regions' coordinates and robot kinematic data, from ROS, are then synced together using *ApproximateTimeSynchronizer* class provided by ROS. The transformations described in Equation 5.6 are then applied to pixel coordinates before publishing them to Rviz using the *PointCloud* format.
4. Upon completion of ultrasound scanning, the global coordinates of the 3D point cloud visualization are sent to the optimal insertion planning algorithm described in section 5.3.5. Outputs from the algorithm regarding the arterial bifurcation and ligament landmarks and insertion scores are then published to Rviz for display.

All of the above steps are completed with zero human intervention. A diagram of the system is displayed in Figure 5.2.

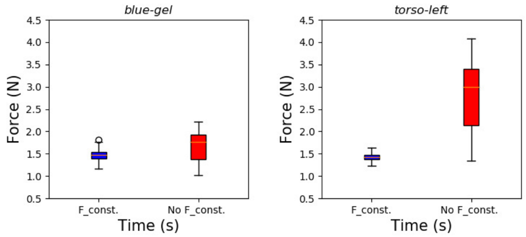


Figure 5.6: Boxplots displaying the mean and standard deviation of the force response values with the force constraint on (left) and off (right). This illustrates the stability of our contact force due to the force constraint.

5.4 Analysis and Results

5.4.1 Robot Controller

We evaluate the performance of our robotic scanning procedure from two different angles: the consistency of the contact force used for scanning and the stability of the ultrasound image outputs.

Stability of Contact Force. Multiple scans were performed across the *blue-gel* and *torso-left* phantoms with the only varying factor being whether the force constraint was turned on or off. The starting desired force in the direction of the probe was 1.5 Newtons (N) for both the on/off variations of the force constraint, except the 1.5 N constraint was removed as soon as scanning started for the latter case. To illustrate the stability of our contact force, we noted the standard deviation values of the applied force for *blue-gel*: $\sigma = 0.113, 0.303$ and for *torso-left*: $\sigma = 0.080, 0.748$ all for force constraint on and off, respectively. For *torso-left*, we noted that, without force constraint, the force continuously rises due to the probe not being able to track the curved *torso-left* surface profile. Figure 5.6 shows the mean and standard deviation values of the aforementioned trials, outlining the consistency of our scanning force methodology.

Stability of Ultrasound Images. To quantify the stability across ultrasound image sequences, we use the normalized cross-correlation (NCC) method [88]. Due to the multiple anatomical classes in the phantoms, the traditional NCC decision to use only the first image in the sequence as the template results in artificially low values. To address that, we use an adaptive form of NCC where image $i - \delta_{gap}$ was used as the template, where i is the current image and δ_{gap} is a parameter for the number of frames in the gap. We decided to take the average NCC metric over $\delta_{gap} \in [10, 30]$, to account for potential image variations within the last second of scanning. For the upper bound of the metric, we evaluated the condition where the robotic arm is locked in a fixed location with constant force. For the lower bound, we evaluated the scenario where the robotic arm force constraint is turned off and is repeatedly making jittery contact with the surface. The experimental results across 3 scans for *blue-gel*, *torso-left* and *torso-right* are shown in Figure 5.7. In **9/9** of the trials, our NCC metric remained closer to the upper bound. We further note that, in all our experiments, we did not notice any vessel deformation or transducer slippage due to our

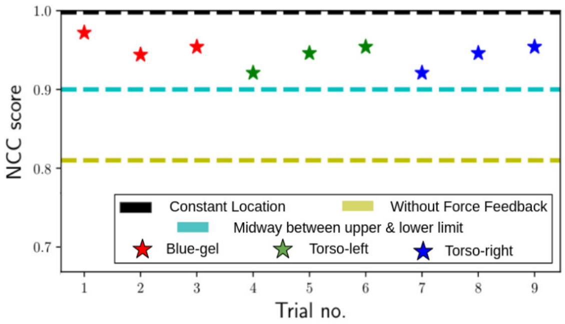


Figure 5.7: Our adaptive NCC score for different ultrasound scanning scenarios. For all of our trials, the stability of the scanned ultrasound images approached the upper bound of a smooth scan.

controller implementation. The force applied is typical for the ultrasound domain [79, 89]. Exploring effects of scanning on human tissue is an interesting path for future work.

5.4.2 Multi-Class Segmentation

To validate the robustness of our multi-class segmentation network, we evaluate its performance on *blue-gel*, *torso-left*, *torso-right*, and on the *live-pig* datasets, all of which (except for *live-pig* which was just with im_a) are compared across 3 varying imaging settings: (im_a) depth of 5 cm and gain value of 15 units, (im_b) depth of 10 cm and gain value of 15 units, and (im_c) a depth of 10 cm with a gain value of 10 units. Each of the phantom datasets contained a train/valid/test split of 320/128/256 images, whereas the live-pig dataset contained a split of 640/320/320 images. The images were split into sequential groups of 8 frames each. After applying data augmentation to the training images, as also described in [90], each set of images increased by a factor of ~ 20. The outputs from the final layers were converted to color encoded masks using a threshold value of 0.50. The erosion morphological operation is used to convert the dense segmentation mask into just the border of the circle. Samples of the images are shown in Figure 5.5.

For baseline comparisons, we evaluated the results against those of a vanilla 3D U-Net, which we refer to as $3DU$, [69] and a different variation of the Bayesian 3D U-Net similar to that in [71], which we refer to as $B3DU_k$. The metrics we used are region similarity, J, contour accuracy, F, and temporal stability, T [81]. We evaluate the spatial region and contour segmentation performance of estimated segmentation, S and ground-truth mask, G, using region similarity and contour accuracy, and the stability and jitteriness across the temporal domain using temporal stability. We calculate the metrics as described in [91], where region similarity is the *intersection-over-union* between S and G, contour accuracy is the F-measure over the precision and recall of the contour points between S and G, and temporal stability is the resulting mean cost per matched point from the Dynamic Time Warping problem [91, 92]. The multi-class segmentation model performances are described in Table 5.1. As can be seen, *our model always obtained the most accurate boundary contours* and our model often significantly outperformed the other state-of-the-

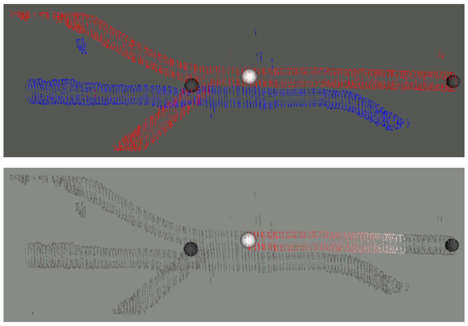


Figure 5.8: 3D visualization of *blue-gel*. The optimal insertion point is the white dot, whereas the ligament and arterial bifurcation points are the grey dots. The image on the bottom illustrates a heatmap of the *Total Site Scores* for the needle planning algorithm (unsafe to safe goes from gray to red).

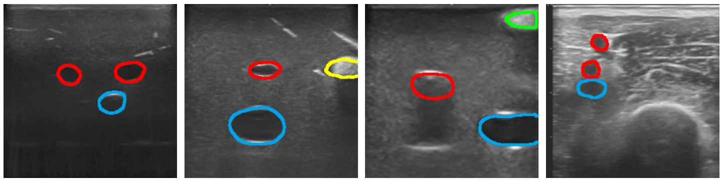


Figure 5.9: Segmentation results on images collected with different imaging settings and anatomy. Left to right: *blue-gel*, *torso-left*, *torso-right*, *live-pig*.

art networks for this sequential medical imaging segmentation task. Sample segmentation outputs are shown in Figure 5.9.

5.4.3 Optimal Needle Insertion Planning

To test for robustness, 9 trials of 3D visualization and the optimal needle insertion algorithm were repeated across *blue-gel*, *torso-left*, and *torso-right*, with each one from imaging settings im_a, im_b, and im_c - described in section 5.4.2. We also completed 8 trials for different sequences of *live-pig*, which is most similar to reality. We assumed a venous insertion in *torso-left* and *torso-right* due to anatomical differences in the phantoms, whereas *blue-gel* and *live-pig* remained as arterial insertions. For *live-pig*, we did have to tune k, the gap of arterial contours for the endpoints of $\overrightarrow{u}$ and $\overrightarrow{v}$ (Figure 5.11), within the range of 1 and 5.

To evaluate the clinical viability of our proposed insertion planning algorithm, we asked 3 doctors to judge the accuracy of the results. We initially only showed the doctors the naked 3D visualizations, without any insertion points shown, and asked them to make unbiased judgments of safe regions for arterial/venous insertion along with their respective vessel bifurcation and ligament points. We then presented to them the 3D visualizations and heatmaps, with optimal insertion, bifurcation, and ligament points shown, and asked for them to confirm their correctness. On *blue-gel*, *torso-left*, and *torso-right*, our algorithm determined the proper anatomical landmarks and insertion points **9/9** times. On *live-pig*,

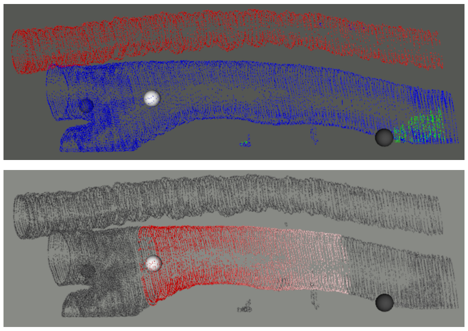


Figure 5.10: 3D visualization of *torso-right*. Optimal insertion point is the white dot, whereas the ligament and arterial bifurcation points are the grey dots. The image on the bottom illustrates a heatmap of the *Total Site Scores* for the needle planning algorithm (unsafe to safe goes from gray to red).

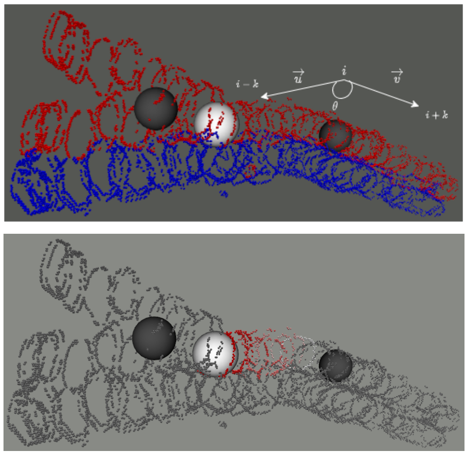


Figure 5.11: 3D visualization of *live-pig*. The top image shows the method we use for detecting the ligament. The optimal insertion point is the white dot, whereas the ligament and arterial bifurcation points are the grey dots. The image on the bottom illustrates a heatmap of the *Total Site Scores* for the needle planning algorithm (unsafe to safe goes from gray to red).

Table 5.1: Evaluation with region similarity (J), contour accuracy (F), and temporal stability (T). Arrows indicate optimal direction for avg ± std across im_a, im_b, im_c (im_a for *live-pig*).

.

Model	$J \uparrow$	$F \uparrow$	$T \downarrow$
$blue$-gel			
$3DU$	.775 ± .035	.291 ± .050	**.085** ± **.083**
$B3DU_k$	.785 ± .015	.316 ± .064	.122 ± .113
$Ours$	**.834** ± **.008**	**.516** ± **.008**	.098 ± .078
$torso$-$left$			
$3DU$	.663 ± .014	.442 ± .025	.169 ± .060
$B3DU_k$	**.793** ± **.107**	.656 ± .140	.131 ± .052
$Ours$	.788 ± .114	**.662** ± **.140**	**.124** ± **.033**
$torso$-$right$			
$3DU$	.706 ± .076	.402 ± .007	**.139** ± **.023**
$B3DU_k$	.807 ± .041	.571 ± .032	.192 ± .033
$Ours$	**.816** ± **.057**	**.657** ± **.013**	.151 ± .045
$live$-pig			
$3DU$	.497 ± .081	.294 ± .011	.092 ± .043
$B3DU_k$	.667 ± .061	.482 ± .057	.065 ± .024
$Ours$	**.814** ± **.065**	**.681** ± **.055**	**.061** ± **.012**

our algorithm correctly determined the ligament **8/8** times, the arterial bifurcation **7/8** times, and a safe insertion point **6/8** times. The 2 sequences without a safe insertion point were instead deemed slightly too close to the arterial bifurcation. Overall, our algorithm detected the proper arterial bifurcation and ligament landmarks **31/32** times, while detecting a safe insertion point **15/17** times. For additional validation, we also displayed plots illustrating the range and general distribution of proximity hazard scores, overlap scores, and total site scores. Computationally, just the insertion planning algorithm takes ˜0.0500 seconds to complete, averaged over 10 runs. Overall, a single-direction scan on the entire robotic system takes about 18-19 seconds on the lower torso phantom, with the 3D visualization running in parallel. Figures 5.1, 5.8, 5.10, and 5.11 illustrate our results.

5.5 Conclusion and Future Work

We present a novel fully automatic robotic system for planning safer needle insertion into the femoral vessels based on 3D visualizations built from validated scanning and multi-class segmentation, all within a Bayesian framework. Through the use of the model, arteries/veins and ligaments can be accurately used as anatomical landmarks for guiding needle insertion. This frees up the medical clinician from the cognitive burden of determining them manually and remembering where they are within the 3D space of the patient, all during tense emergency scenarios. With the 3D visualization and novel needle insertion algorithm, the user is also able to gain clearly explainable results for understanding. All of this is completed with no human intervention and can be easily portable and generalizable to other locations and anatomies, as shown by our extensive validation across several different anatomical variations and imaging settings - resulting in safe insertion locations 88% of the time. As a result, this is the first work to potentially significantly decrease the amount of medical training necessary for performing emergency medical deep-vessel insertion operations.

In the future, we aim to increase the portability of our robotic systems and add new capabilities for physical insertion of the needle. We also plan to develop for scanning highly curved surfaces with continuously changing surface normals, along with increase the segmentation and needle insertion algorithm's robustness to other datasets.

Chapter 6

Conclusion

This thesis explored research problems involving the primary challenges of ultrasound images, which are often intertwined with each other: (1) their immense variability across scanners, imaging settings, scanning patterns, and body types, and (2) the high costs associated with obtaining training data. Chapter 2 aimed to study these challenges in-depth from the perspective of transfer learning, a method often used when training data for the target task is lacking. Traditional transfer learning strategies often fine-tune the entire network with a smaller set of training data, with the goal being to leverage a prior set of weights for enhanced downstream performance and generalizability. However, in chapter 2, we studied different variations of the fine-tuning strategy, applied to various contiguous subsequences of the U-Net model architecture. The work further evaluated such fine-tuning strategies on multiple unseen test datasets as a test for their average out-of-training-domain performance. Upon the studies, we proposed to use the fine-tune data domain's performance as a proxy for the strategy's out-of-training-domain generalizability score, allowing empirical determination of *which specific blocks* should and should not be fine tuned to maximize expected generalization. We find that this results in significantly enhanced transfer learning generalization when compared to previous methods.

Chapters 3 and 4 focused on addressing the issues of data variability and shortage using a method known as data augmentation, where the goal is to generate synthetic copies of the data to enlarge the training set. Chapter 3 explicitly focused on generalizing across various ultrasound scanning patterns - so in the temporal sense. To do so, we introduced three different stochastic temporal augmentation strategies, each of which used stochasticity to create time-varying ultrasound frame sequences from the original set of data. Chapter 4 more directly addressed the spatial variability across body types, injury scenarios, and imaging artifacts. In doing so, we introduced an uncertainty-based, online adaptive data augmentation method which produces non-uniform spatial distortions within each ultrasound image. By using epistemic uncertainty maps to understand the current model strengths, we were able to show that such an augmentation can greatly help the generalizability of the downstream segmentation model.

Chapter 5 introduced an initial prototype for an automatic robotic system for needle insertion, demonstrating how accurate semantic segmentation of ultrasound images can result in proper determination of the optimal needle insertion location. We introduced how we set up the overall robotic pipeline, consisting of automated ultrasound scanning, ultrasound segmentation, 3D model reconstruction, and then optimal needle insertion location determination. We evaluated our system on both phantom and live-pig data and noticed the majority of the optimal needle insertion locations being determined properly.

In the future, I hope to extend each of the above works further. Although fairly simple, I believe transfer learning provides great potential for leveraging prior large sets of data for enhanced generalizability in small-data scenarios. In the future, I believe it may be useful to further study how generalization and transfer learning varies across ultrasound data, or to potentially combine it with other methods such as meta-learning.

While a large portion of similar research is focused on finding enhanced model architectures, I also believe there should be a great amount of focus on the data itself. Although I have introduced a couple data augmentation methods to combat generalization above, I still believe there is significant work in this area. Much work can be done to reduce computational and memory requirements of online data augmentation methods or to further understand how they are affecting generalization. Harmful data augmentations are also possible, so it would be very useful to systematically be able to check for that and understand how it affects the model out-of-training-domain performance. Overall, I believe there is immense potential in enhancing the generalizability of ultrasound AI methods and am excited for the future it allows.

Bibliography

[1] C. E. Jun. Pictorial essay ultrasonographic evaluation of complications related to transfemoral arterial procedures. *e-ultrasonography*, 2020. (document), 5.4

[2] J. A. Jensen. Medical ultrasound imaging. *Progress in Biophysics and Molecular Biology*, 2007. 1.1

[3] T.S. Mathai, V. Gorantla, and J. Galeotti. Segmentation of vessels in ultra high frequency ultrasound sequences using contextual memory. In *International Conference on medical image computing and computer-assisted intervention*, 2019. 1.2, 2.2

[4] T.S. Mathai, L. Jin, V. Gorantla, and J. Galeotti. Fast vessel segmentation and tracking in ultra high-frequency ultrasound images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2018. 1.2, 2.2

[5] J. Parker. System and method for managing medical data, 2006. 2.1

[6] S. Mani, A. Sankaran, S. Tamilselvam, and A. Sethi. Coverage testing of deep learning models using dataset characterization. *arXiV*, 2019. 2.1

[7] A. Chen, M. Balter, T. Maguire, and M. Yarmush. Deep learning robotic guidance for autonomous vascular access. *Nature Machine Intelligence*, 2020. 2.1

[8] O. Dudeck, U. Teichgraeber, P. Podrabsky, E. Haenninen, R. Soerensen, and J. Ricke. A randomized trial assessing the value of ultrasound-guided puncture of the femoral artery for interventional investigations. *The International Journal of Cardiovascular Imaging*, 2004. 2.1

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015. 2.1

[10] R. Geirhos, P. Rubish, C. Michaelis, M. Bethge, F. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture: increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. 2.1

[11] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio. Transfusion: Understanding transfer learning for medical imaging. In *Advances in Neural Information Processing Systems*, 2019. 2.1

[12] V. Iglovikov and A. Shvletz. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiV*, 2018. 2.1

[13] D. Pakhomav, V. Premachandran, M. Allan, M. Azizian, and N. Navab. Deep residual

learning for instrument segmentation in robotic surgery. In *International Workshop on Machine Learning in Medical Imaging*, 2019. 2.1

[14] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on medical image computing and computer-assisted intervention*, 2015. 2.1

[15] M. Amiri, R. Brooks, and H. Rivaz. Fine tuning u-net for ultrasound image segmentation: which layers? In *International conference on medical image computing and computer-assisted intervention*, 2019. 2.1

[16] V. Patel. A framework for secure and decentralized sharing of medical imaging data via blockchain consensus. *Health Informatics Journal*, 2019. 2.1

[17] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. Tensorflow: A system for large-scale machine learning. In *USENIX Symposium on Operating Systems Design and Implementation*, 2016. 2.2

[18] D. Kingma and J. Ba. Adam: A method for stochastic optimization. 2014. 2.2

[19] Y. Xia, X. Cao, F. Wen, G. Hua, and J. Sun. Learning discriminative reconstructions for unsupervised outlier removal. In *International Conference on Computer Vision*, 2015. 2.3, 2.3.3

[20] Y. Amatya, J. Rupp, F. M. Russell, J. Saunders, B. Bales, and D. R. House. Diagnostic use of lung ultrasound compared to chest radiograph for suspected pneumonia in a resource-limited setting. *International Journal of Emergency Medicine*, 11, 2018. 3.1

[21] T. Ogura, A. K. Lefor, M. Nakamura, K. Fujizuka, K. Shiroto, and M. Nakano. Ultrasound-guided resuscitative endovascular balloon occlusion of the aorta in the resuscitation area. *The Journal of Emergency Medicine*, 52, 2017. 3.1

[22] E. Smistad and F. Lindseth. Real-time automatic artery segmentation, reconstruction and registration for ultrasound-guided regional anaesthesia of the femoral nerve. *IEEE Transactions on Medical Imaging*, 2015. 3.1

[23] E. Smistad and L. Løvstakken. Vessel detection in ultrasound images using deep convolutional neural networks. *Deep Learning and Data Labeling for Medical Applications*, 2016.

[24] E. Smistad, D. H. Iversen, L. Leidig, J. B. L. Bakeng, K. F. Johansen, and F. Lindseth. Automatic segmentation and probe guidance for real-time assistance of ultrasound-guided femoral nerve blocks. *Ultrasound in medicine & biology*, 2017. 3.1

[25] W. A. Teeter, J. Matsumoto, K. Idoguchi, Y. Kon, T. Orita, T. Funabiki, M. L. Brenner, and Y. Matsumara. Smaller introducer sheaths for reboa may be associated with fewer complications. *Journal of Trauma and Acute Care Surgery*, 2016. 3.1

[26] A. Zaman, S. H. Park, H. Bang, C. Park, I. Park, and S. Joung. Generative approach for data augmentation for deep learning-based bone surface segmentation from ultrasound images. *International Journal of Computer Assisted Radiology and Surgery*, 2020. 3.1

[27] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy. Deep learning approaches for data augmentation and classification of breast masses using ultrasound images. *Int. J. Adv. Comput. Sci. Appl*, 2019. 3.1

[28] J. Tu, H. Liu, F. Meng, M. Liu, and R. Ding. Spatial-temporal data augmentation based on lstm autoencoder network for skeleton-based human action recognition. In *IEEE International Conference on Image Processing*, 2018. 3.1

[29] A. L. Guennec, S. Malinowski, and R. Tavenard. Data augmentation for time series classification using convolutional neural networks. 2016. 3.1, 3.2, 3.3, 3.4

[30] S. Haradal, H. Hayashi, and S. Uchida. Biosignal data augmentation based on generative adversarial networks. In *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2018.

[31] A. I. Chen, M. Balter, T. J. Maguire, and M. L. Yarmush. Deep learning robotic guidance for autonomous vascular access. *Nature Machine Intelligence*, 2020. 3.1

[32] O. Çiçek et al. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, 2016. 3.3, 4.2.1

[33] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiV*, 2014. 3.3

[34] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 3.3

[35] C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 2019. 3.4

[36] H. Moritz, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, 2016. 3.4

[37] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. *arXiV*, 2016.

[38] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017. 3.4

[39] F. Milletari, N. Navab, and S. Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *International Conference on 3D Vision*, 2016. 4.1

[40] Z. Zhou et al. Unet++: A nested u-net architecture for medical image segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, 2018.

[41] L. Chen et al. Attention unet++: A nested attention-aware u-net for liver ct image segmentation. In *IEEE International Conference on Image Processing*, 2020. 4.1

[42] C. Shorten and T. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 2019. 4.1

[43] E. Cubuk et al. Autoaugment: Learning augmentation policies from data. *arXiV*, 2018. 4.1

[44] E. Cubuk et al. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020. 4.1

[45] C. Luo et al. Learn to augment: Joint data augmentation and network optimization for text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 4.1, 4.2.2

[46] S. Lim et al. Fast autoaugment. In *Advances in Neural Information Processing Systems*, 2019. 4.1

[47] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, 2016. 4.1, 4.2.1

[48] Y. Gal. Uncertainty in deep learning. *PhD Thesis*, 2016.

[49] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, 2017. 4.1, 4.2.1, 4.2.1, 4.2.1, 4.3.2

[50] S. Schaefer, T. McPhail, and J. Warren. Image deformation using moving least squares. In *ACM SIGGRAPH 2006 Papers*, 2006. 4.1, 4.2.2, 4.2.2, 4.2.2, 4.2.2, 4.4

[51] Y. Kwon et al. Uncertainty quantification using bayesian neural networks in classification: Application to biomedical image segmentation. *Computational Statistics and Data Analysis 142*, 2020. 4.2.1

[52] J. Lee and H. Byun. Image deformation using moving least squares based on unequal grids. *International Journal of Software Engineering & Its Applications 6*, 2013. 4.2.2, 4.2.2

[53] F. Perazzi et al. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 4.3.4

[54] J. Y. Tsui, A. B. Collins, D. W. White, J. Lai, and J. A. Tabas. Placement of a femoral venous catheter. 2008. 5.1

[55] Percutaneous coronary intervention: A report from the national cardiovascular data registry. *JACC: Cardiovascular Interventions, Volume 1, Issue 4, August 2008*, 2008. 5.1

[56] F. J. Criado, O. Abul-Khoudoud, and E. Wellons. Percutaneous femoral puncture for endovascular treatment of occlusive arterial lesions. *Am J Surg*, 1998. 5.1

[57] O. Dudeck, U. Teichgraeber, P. Podrabsky, E. L. Haenninen, R. Soerensen, and J. Ricke. A randomized trial assessing the value of ultrasound-guided puncture of the femoral artery for interventional investigations. *The International Journal of Cardiovascular Imaging*, 2004. 5.1

[58] B. M. Snelling, S. Sur, S. S. Shah, M. M. Marlow, M. G. Cohen, and E. C. Peterson. Transradial access: lessons learned from cardiology. *Journal of NeuroInterventional Surgery, Volume 10, Issue 5*. 5.1

[59] A. Chen et al. Deep learning robotic guidance for autonomous vascular access. *Nature Machine Intelligence*, 2020. 5.1, 5.2

[60] D. Castro, L. Lee, and B. Bhutta. Femoral vein central venous access. *StatPearls Publishing*, 2021. 5.1

[61] E. England, C. Spear, D. Huang, J. Weinberg, J. Bogert, T. Gillespie, and J. Mankin. Reboa as a rescue strategy for catastrophic vascular injury during robotic surgery. *Journal of Robotic Surgery*, 2020. 5.1

[62] M. Cilingiroglu, T. Feldman, M. H. Salinger, J. Levisay, and Z. G. Turi. Fluoroscopically-guided micropuncture femoral artery access for large-caliber sheath insertion. *JIC Volume 23 - Issue 4*, 2011. 5.2

[63] J. Jayender, M. Azizian, and R. V. Patel. Autonomous image-guided robot-assisted active catheter insertion. *IEEE Transactions on Robotics*, 2008. 5.2

[64] E. Smistad, D. H. Iversen, L. Leidig, J. B. L. Bakeng, K. F. Johansen, and F. Lindseth. Automatic segmentation and probe guidance for real-time assistance of ultrasound-guided femoral nerve blocks. *Ultrasound in Medicine*, 2017. 5.2

[65] M. Vogt and D. J. van Gerwen. Optimal point of insertion of the needle in neuraxial blockade using a midline approach: study in a geometrical model. *Local Reg Anesth.*, 2016. 5.2

[66] T. T. Oh, M. Ikhsan, K. K. Tan, S. Rehena, N. R. Han, A. T. H. Sia, and B. L. Sng. A novel approach to neuraxial anesthetia: application of an automated ultrasound spinal landmark identification. *BMC Anesthesiology*, 2019. 5.2

[67] D. Castro, L. Martin Lee, and B. Bhutta. Femoral vein central venous access. *StatPearls*, 2020. 5.2

[68] O. Ronneberger, P. Fisher, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 5.2

[69] O. Cicek et al. 3d u-net: Learning dense volumetric segmentation from sparse annotation. In *MICCAI*, 2016. 5.2, 5.3.3, 5.4.2

[70] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*, 2017. 5.2, 5.3.3, 5.3.3, 5.3.3

[71] Y. Kwon, J. Won, B. Kim, and M. Paik. Uncertainty quantification using bayesian neural networks in classification: Application to ischemic stroke lesion segmentation. In *MIDL*, 2018. 5.2, 5.4.2

[72] E. Smistad and F. Lindseth. Real-time automatic artery segmentation, reconstruction, and registration for ultrasound-guided regional anaesthesia of the femoral nerve. *IEEE Transactions on Medical Imaging*, 2016. 5.2

[73] O. Hadjerci, A. Hafiane, P. Makris, D. Conte, P. Vieyres, and A. Delbos. Nerve detection in ultrasound image using median gabor binary pattern. *Image Analysis and*

Recognition, 2014. 5.2

[74] S. Bangalor and D. L. Bhatt. Femoral arterial access and closure. *American Heart Association*, 2011. 5.2

[75] F. Burzotta, O. Shoeib, C. Aurigemma, and C. Trani. Angio-guidewire-ultrasound (agu) guidance for femoral access in procedures requiring large sheaths. *Journal of Invasive Cardiology*, 2019. 5.2

[76] M. Abadi et al. Tensorflow: Large-scale machine learning on heterogenous distributed systems. *12th USENIX Symnposium on Operating Systems Design and Implementation*, 2016. 5.3.1

[77] M. Quigley et al. Ros: An open-source robot operating system. In *ICRA Workshop on Open Source Software*, 2009. 5.3.1

[78] R. Finocchi et al. 3d co-robotic ultrasound imaging: a cooperative force control approach. *Medical Imaging*, 2017. 5.3.2

[79] M. Victorova, D. Navarro-Alarcon, and Y. Zheng. 3d ultrasound imaging of scoliosis with force sensitive robotic scanning. 2019. 5.3.2, 5.4.1

[80] D. P. Kingma and J. Ba et al. A method for stochastic optimization. In *ICLR*, 2015. 5.3.3

[81] C. E. Clark. The pert model for the distribution of an activity time. *Operations Research*, 1962. 5.3.4, 5.4.2

[82] L. Mercier. Review of ultrasound probe calibration techniques for 3d ultrasound. 2004. 5.3.4

[83] Z. G. Turi. An evidence-based approach to femoral arterial access and closure. *Reviews in Cardiovascular Medicine Vol. 9 No. 1*, 2008. 5.3.5

[84] S. B. Rupp. Relationship of the inguinal ligament to pelvic radiographic landmarks: anatomic correlation and its role in femoral arteriography. *J. Vasc. Interv. Radiology*, 1993. 5.3.5

[85] P. Gopalakrishnan et al. Mid femoral head is not the ideal fluroscopic landmark for common femoral artery puncture. *Journal of the American College of Cardiology*, 2017. 5.3.5

[86] L. Oğuzkurt et al. Ultrasound-guided puncture of the femoral artery for total percutaneous aortic aneurysm repair. *Interventional Radiology, Volume 18 Issue 1*, 2011. 5.3.5

[87] P. Garrett, R. Eckart, T. Bauch, C. Thompson, and K. Stajduhar. Fluoroscopic localization of the femoral head as a landmark for common femoral artery cannulation. *Catheter Cardiovasc Interv.*, 2005. 5.3.5

[88] T. Y. Fang, H. K. Zhang, R. Finocchi, R. H. Taylor, and E. M. Boctor. Force-assisted ultrasound imaging system through dual force sensing and admittance robot control. *Int J Comput Assist Radiol Surg*, 2017. 5.4.1

[89] K. Mathiassen, J. Fjellin, K. Glette, P. Hol, and O. Elle. An ultrasound robotic system using the commercial robot ur5. *Frontiers in Robotics and AI*, 2016. 5.4.1

[90] T. S. Mathai, V. Gorantla, and J. Galeotti. Segmentation of vessels in ultra high frequency ultrasound sequences using contextual memory. In *MICCAI*, 2019. 5.4.2

[91] F. Perazzi et al. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 5.4.2

[92] L. Rabiner and B. H. Juang. Fundamentals of speech recognition. 1993. 5.4.2