KNEE CARTILAGE SEGMENTATION OF ULTRASOUND IMAGES USING CONVOLUTIONAL NEURAL NETWORKS AND LOCAL PHASE ENHANCEMENT

BY JUSTIN HEERALALL MOHABIR

A thesis submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Master of Science

Graduate Program in Biomedical Engineering

Written under the direction of

Ilker Hacihaliloglu

And approved by

New Brunswick, New Jersey
May 2020

ABSTRACT OF THE THESIS

KNEE CARTILAGE SEGMENTATION OF ULTRASOUND IMAGES USING CONVOLUTIONAL NEURAL NETWORKS AND LOCAL PHASE ENHANCEMENT

By Justin Heeralall Mohabir
Thesis Director: Ilker Hacihaliloglu

Osteoarthritis (OA) is a chronic disorder that results from the inflammation of body joints and the degradation of cartilage. The most prominent form of OA is knee OA, where the cartilage between the femur and tibia degrades from regular use. To measure the progression of knee OA in patients, clinicians use a metric of cartilage thickness known as Joint Space Width (JSW) to see how much cartilage is degraded over time. The most common method of measuring JSW is to perform a planar X-ray on the knee and manually measure the space between the joints from that image. This, however, gives patients a dose of ionizing radiation. Magnetic Resonance (MR) imaging and Ultrasound (US) have arisen as alternatives to imaging knee cartilage. MR imaging is reserved to research settings due to the expensive operation. This leaves US as the main alternative to show promise from clinical studies but has limitations such as noise and artifacts that make segmentation of the knee cartilage within images difficult to segment manually. A previous study has shown that enhancing images prior to segmentation can allow a more accurate segmentation. This thesis investigated the efficacy of using different Convolutional Neural Network (CNN) architectures to segment knee cartilage from US images, as well as the effect of enhancing the images prior to segmentation from the CNNs compared to a Random Walker (RW) algorithm.

The CNN architectures used in this study are: U-Net, Stacked U-Net and W-Net. Each of these architectures were trained by either B-mode images, local phase enhanced images, or an early-stage combination of both the B-mode and enhanced images. The 150-image training set of data used was augmented to artificially increase the amount of training images to improve the robustness and to prevent overfitting. 10-fold cross-validation was performed on each combination of CNN architecture and input type to prevent outliers. Validation was performed on each of the CNNs generated by comparison against a manual segmentation of the US images using the Dice Similarity Coefficient (DSC). Validation was performed on 50 images from a similar dataset used to train the CNNs and a second set of 50 images from a different US system. The average DSC for the U-Net, Stacked U-Net and W-Net were: 0.8566, 0.8289 and 0.8675 in the similar dataset and 0.779, 0.7185 and 0.772 in the different dataset, respectively. The average DSC for the B-Mode, enhanced, and combined input types were: 0.8071, 0.8552 and 0.8908 in the similar dataset and 0.6869, 0.7756 and 0.807 in the different dataset, respectively. Compared to a RW algorithm, 53% of U-Nets, 67% of Stacked U-Nets, and 70% of W-Nets had significantly ($p>0.05$) higher average DSCs. 30% of B-Mode networks, 77% of enhanced image networks and 83% combined image networks had significantly higher DSCs. This study presents an automated US cartilage segmentation method using CNNs. The results presented show significant improvements in segmentation using local phase enhancement instead of an unaltered B-Mode US image. Low segmentation time and processing requirement of CNNs show promise as a method of achieving accurate real-time segmentation of knee cartilage and can make US a viable alternative to X-ray for diagnosis and progression measurement of knee OA.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Thesis Motivation

Osteoarthritis (OA) is the degradation of bone joints as a result of frequent or heavy use over time. OA is most seen in hand, knee and spine joints and has a higher prevalence in the elderly. Over time, however, knee OA has seen a gradual increase in populations following the industrial revolution, skyrocketing from 6% to 16% in people over the age of 50 (Wallace, 2017). This is mostly accredited to the rise of obesity and decline of activity in developed populations. As a result of obesity as a result of knee OA, patients can develop coronary heart disease, hypertension and diabetes (Zanella, 2001). The effects of knee OA can be seen with the correlations between knee OA and depression, pain, and an overall lower of quality of life (Segal, 2015).

The effects of knee OA are not only biological; OA has cost adult victims in the US an average of $3952 a year and elderly victims an average of $5704 a year in direct costs alone (Bitton, 2009). The indirect costs of knee OA averaged around $1700 and accounts for losses in wage and productivity. In severe late stages of OA, a Total Knee Replacement (TKR) is usually recommended for patients and can lead to costly surgery and revisions that continue the economic burden on the victims even further. Early diagnosis of knee OA can lead to better, more effective treatment for patients and can overall increase the quality of life of patients.

1.2 Knee Cartilage Anatomy

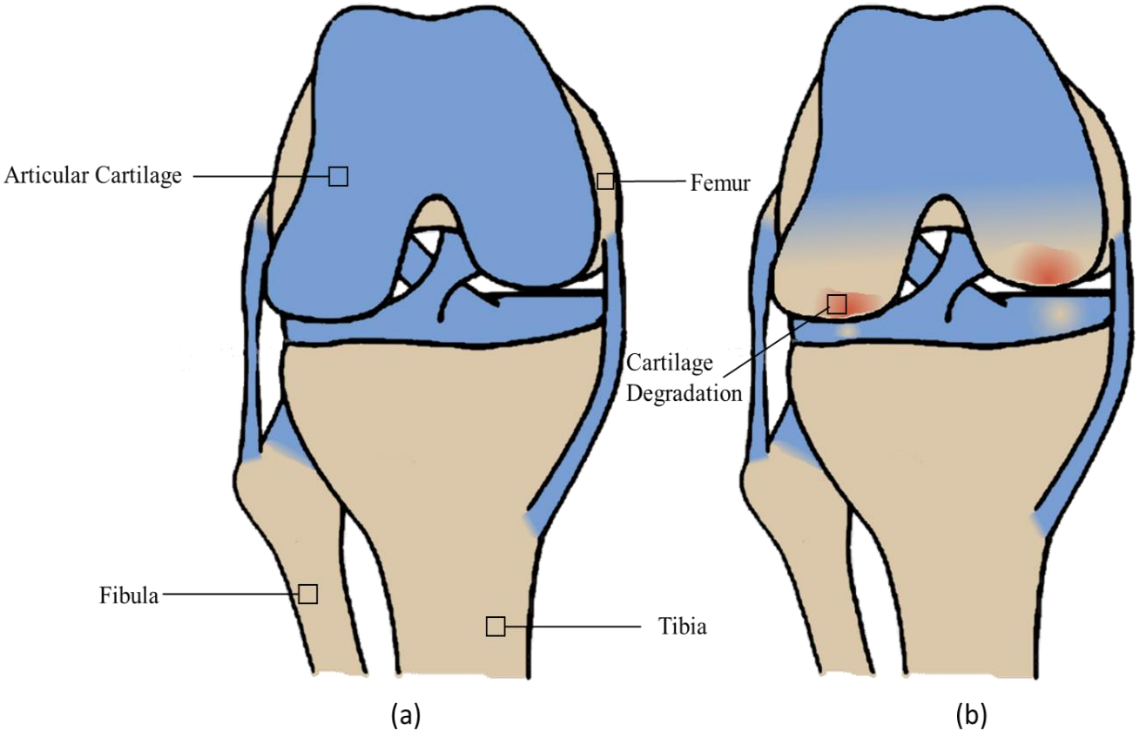


Figure 1.1: An anatomical representation of a healthy knee joint (a) and a knee joint with osteoarthritis (b). The fibrous articular cartilage is shown in blue and caps over the femur and tibia bones in beige and is prone to degradation in those with osteoarthritis, possibly exposing bone and causing pain shown in red.

The primary joint in the knee is located between the femur and tibia. This joint is covered by white fibrous tissue called articular cartilage. The main function of this tissue is to protect the two bones from the wear of rubbing along each other. This cartilage covers the distal end of the femur and the proximal end of the tibia. The patella, which lies on the ventral side of the joint, is also covered in articular cartilage on the dorsal side of the bone. The articular cartilage forms two structures on the tibia named the lateral and medial meniscus. These structures are visualized in Figure 1.1. These structures function to absorb the shock from impact on the joint.

Over time, the cartilage structures in the knee start to degrade as a result of fatigue and age on the articular surfaces, called osteoarthritis (OA). Cartilage degradation leads to pain from direct or near-direct contact of the bones on each other. The degradation also leads to clustering of chondrocytes and bone spur formation that can protrude into the joint. The combination of these effects can lead to limited range of motion and chronic pain in patients and an overall lower quality of life. There are many different treatments available for OA and include: surgical intervention, supportive intervention and pharmacological intervention.

1.3. Diagnosis

In order to effectively treat OA, early detection of the underlying cause and degree of OA is key. The primary change in anatomy in cases of OA is the degradation of the articular cartilage. In order to quantify the degradation of cartilage, the joint space width (JSW) is measured. A smaller JSW would indicate there is less cartilage between the knee joint and a farther progressed OA. Traditionally JSW is measured with a planar X-ray and progression of OA is measured with annual JSW measurements (Lepuesne, 1994). This progression becomes more difficult to accurately measure as the JSW narrows in late stage OA. This makes X-ray less accurate in cases of late stage OA. The progression of OA is what is generally used for diagnosis, with many different metrics being used today (Kohn, 2016). In order to view the progression of OA, there is a need to perform multiple X-ray scans on each patient. The need for multiple scans introduces more radiation to patients, limiting the scan time to once a year to limit exposure to patients. The most common metric is the Kellegren-Lawrence (KL) scale which diagnoses based on the presence of Joint Space Narrowing (JSN) and osteophyte development around cartilage (Kellgren, 1957; Schiphof, 2008). Other metrics include: the International Knee Documentation Committee metric which only looks for JSW measurements smaller than 4 mm to be considered OA(Hefti, 1993; Irrgang, 2006), the Brandt and Fairbank metrics which look for bone deformities for lower OA and JSN for higher OA ratings (Fairbank, 1948; Brandt, 1991), and the Jäger-Wirth metrics which only look for arthrosis, a classification for non-immune system degradation of cartilage and joints (Scheller, 2001; Schroeder-Boersch, 1998). Recent developments in various modalities have opened this measurement to Optical Coherence Tomography (OCT), Magnetic Resonance Imaging (MRI) and finally Ultrasound (US) to lower the exposure of patients to ionizing radiation from X-rays while allowing more accurate measurements. Although the former is only used in research because of the expense of the modalities, US exists as a less expensive method of measuring JSW while maintaining the lack of ionizing radiation. Reducing the price per image of the patient and removing the limiting factor of radiation can allow for more scans to be performed and can give a more detailed progression of OA. The ability to also produce multiple images per patient can also give a multi-planar view of the cartilage opposed to the current planar X-ray.

1.4. Literature Review

Convolutional Neural Networks (CNNs) have recently gained popularity in medical image research as an efficient method of segmentation and classification. Multi-layer CNNs have proven to be useful in supervised training tasks to segment regions of interest (ROIs) out of medical scans. The most pervasive and widely used architecture is the U-

Net (Ronnesberger, 2015). Despite the U-Net system being conceived of for some time, Ronneberger et.al popularized the system and prompted many different architectures to be created. These architectures include: 3-D U-net (Çiçek, 2016), V-Net (Milletari, 2016), Stacked U-Nets (Shah, 2018), and W-Net (Chen, 2018) to list a few. The latter two were made to aid in 2-D image segmentation and attempt to overcome different problems in the base U-Net architecture. Much of the work in implementing these architectures are seen in imaging modalities utilizing 3D volumetric slices. In these cases, the benefit to using a CNN is the time saved by automatically scanning through the images compared to manual review for each slice from a person.

1.4.1 X-Ray and MRI

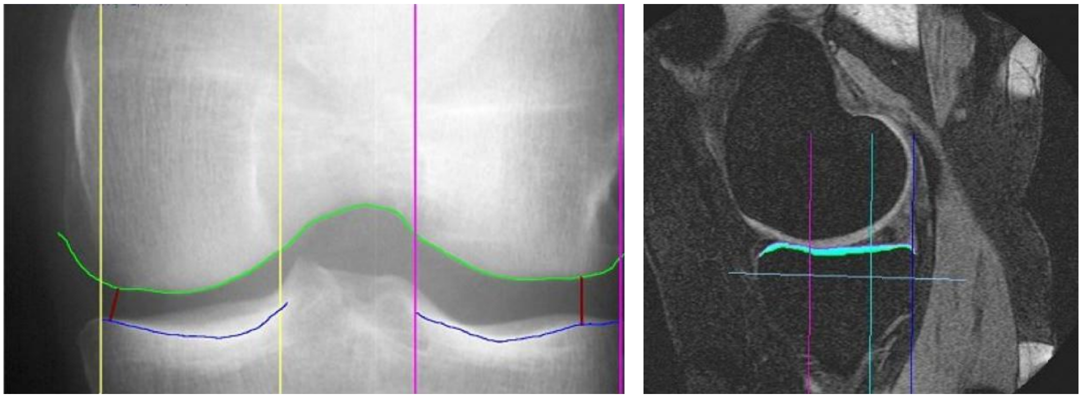


Figure 1.2: A picture showing an anterior X-ray of a knee joint space (left) and a picture generated from a posterior peripheral MR of a knee (right) (Beattie, 2008). Measurement of the JSW in the X-ray is made by connecting the outlines of the bones in green and blue, and measuring that distance as seen in red. The measurement of the JSW in an MRI is done by measuring the soft tissue, shown in cyan.

The current standard for collecting JSW measurements is to use an X-ray image and measure the distance between the high intensity bones. Many automated JSW measurement systems look for bone features in the image to trace the boundary of the bones as shown in the Figure 1.2 (Beattie, 2016; Duryea, 2000). Although X-ray is the most popular method of diagnosing knee OA in clinics, there are few studies looking to automatically segment the joint space from X-ray scans. One study uses a CNN to give a grade on the KL OA scale from the X-ray scan itself (Antony, 2017). This study did not segment the joint space beforehand and used the network to give a numerical value to the X-ray to represent a position on the KL scale. Another similar study uses a Random Forest algorithm to accomplish a similar result (Gornale, 2016). These studies focused more on methods of diagnosis than that of measuring cartilage thickness in the knee.

MR images of the knee measure cartilage by looking for the contrast between the lower intensity bone and higher intensity soft tissue around the bone (Beattie, 2008). An example of an MRI of the knee can be seen in Figure 1.2. MR imaging has emerged as a popular method of measuring JSW in research. The high accuracy and available datasets have provided a platform for automation that X-ray images do not. Proposed algorithms for segmenting knee cartilage include minimizing a locally weighted vote (Lee, 2014), a statistical shape model (Ambellan, 2019), fitting to a Gaussian model (Kashyap, 2016), Random Walkers and Random Forests (Kashyap, 2016; Swanson, 2010; Hong-Seng, 2017; Gornale, 2016) and finally a CNN (Prasoon, 2013). These methods of segmentation are mostly developed to segment knee bone structures along with the joint spaces to reconstruct a 3D model of the cartilage in the knee. This information can give newer insights into the state of a patient’s OA especially for those that plan to receive surgery to alleviate symptoms. Both X-ray and MR have room for automation that can aid in diagnosis accuracy, but both have their disadvantages. X-ray imaging as a modality uses ionizing radiation as a method of imaging, which limits the amount of measurements that can be taken to track progression of OA. MR gets around the problem of radiation but has a different problem in the cost of the imaging devices. X-ray is used in clinical settings and MR is used in research settings for these reasons. This allows Ultrasound to fill this gap by not having ionizing radiation while being relatively inexpensive.

1.4.2 Ultrasound

Ultrasound images of the knee are not typically used for clinically measuring knee cartilage and are instead used in trauma cases to measure damage to soft tissue such as ligaments and tendons (Razek, 2009). The ability of Ultrasound to give real-time feedback on images is crucial in these trauma cases. Ultrasound imaging of knee structures has given room for Neural Networks to lead as a method of segmenting regions of interests. Lower quality images along with noisy artifacts provides a challenge for many simpler segmentation methods such as Random Walker and Watershed (Desai, 2018). Recent works in bone structure segmentations from ultrasound have used multi-feature Convolutional Neural Networks as a method of finding features on the femur and tibia (Wang, 2018). Studies done on cartilage have used various methods of segmentation for different applications. One such example is a study which used a U-Net for segmentation of femoral cartilage to be used as a 3D model for robot-assisted surgeries (Antico, 2020). Another study investigated using an encoder-decoder pair of CNNs to track 3D volumes of cartilage from multiple slices of ultrasound in hopes of making a real-time segmentation algorithm (Dunnhofer, 2020). A final recent study used multiple Neural Networks to first find the location of the knee joint in an image, and then to segment out the femoral cartilage from that new cropped region (Kompella, 2019). This study is particularly interesting in that it uses two independent CNNs to accomplish the segmentation. This allowed for a more robust network that performed well with various untrained datasets, but never reached a significantly high performance within those datasets, maximizing with a Dice Coefficient of 0.80 in images within the training set of images. This gap in Dice coefficient leaves room for improvement of algorithms to achieve low-latency segmentation of femoral cartilage. This can lead to increases in segmentation accuracy, the ability of real-time segmentation with low compute times and the ability to segment at different transducer angles.

1.5. Objective and Scope

The main objective of this thesis is to evaluate the viability of using Convolutional Neural Networks to segment Joint Space Width from US knee cartilage images. Along with this, testing the effect of using image enhancement on the US images and inputs into different CNN structures.

The specific aims are to:

1. The separation and manual segmentation of knee cartilage from two unique datasets of knee ultrasound images.
2. Local Phase filtering enhancement of all images with the same parameters across both datasets.
3. Construction and optimization of three different Convolutional Neural Network architectures: U-Net, Stacked U-Net and W-Net, that are capable of extracting and presenting information about the Joint Space Width.

4. Training each of the three Convolutional Neural network architectures with three different input image types: B-mode images, Enhanced images and early stage fusion of B-mode and Enhanced images.
5. Qualitative and quantitative evaluation of the performance of the trained networks using images reserved for validation compared to the previously established Random Walker algorithm.
6. Investigating the effects of both using three different image types and three architectures.

Chapter 2

Current Methods

2.1 Cartilage in Ultrasound

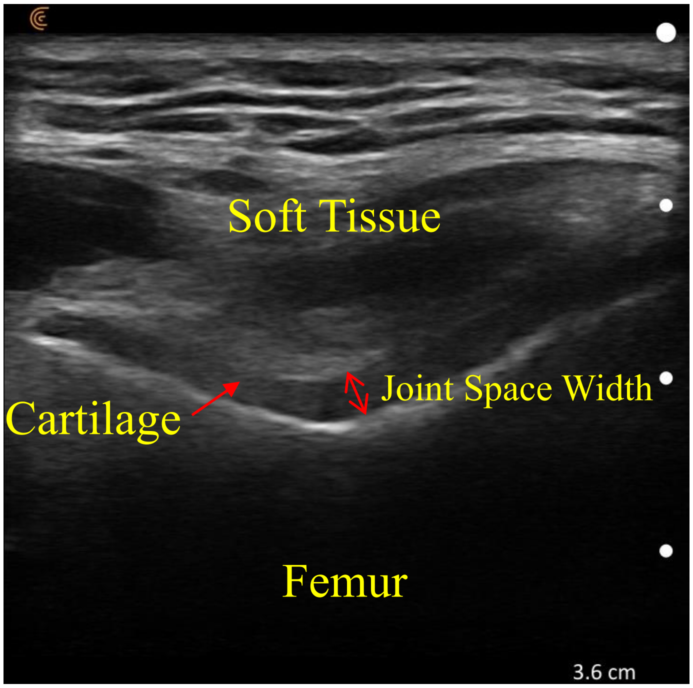


Figure 2.1: An example of ultrasound of a knee with labelled landmarks. This study will focus on segmenting the hypoechoic cartilage band from the ultrasound, as that is what is used to measure the Joint Space Width, corresponding to the thickness of the cartilage. The cartilage is located between the distal edge of the femur and the proximal edge of the tibia.

Ultrasound (US) has been successfully applied for imaging soft tissue for various diagnostic applications (Henderson, 2015). In the case of cartilage, the bone surrounding region of interests (ROI) act as a good contrast in the ultrasound images. Specifically, bone boundaries enable high intensity lighter pixels that show the cartilage as the separation between the bone boundaries as a layer of low intensity darker pixels. These are referred to as hyperechoic (lighter) and hypoechoic (darker) regions of the ultrasound image. In the case of knee ultrasound images, the hyperechoic femur and tibia show the boundaries of the hypoechoic cartilage. The space between these hyperechoic lines on the ultrasound is referred to as the Joint Space Width (JSW) as shown in Figure 2.1 and is the ROI for OA detection.

2.2 Segmentation

2.2.1 Manual Segmentation

The most popular method of measuring the JSW in US knee images is to manually measure the width from the ultrasound image with or without enhancement. This method is cheap, fast and the easiest to integrate into clinical settings, being compatible with most US devices. In studies with large numbers of patients, however, this method can become inefficient. These measurements are usually taken by a US tech but can sometimes require feedback from physicians (Riecke, 2014). This increased overhead in JSW measurement can drive up costs of getting an US scan. The method of measuring JSW manually is to take an image of the B-Mode US image of the knee at an angle of inflexion and to measure from the distal end of the femur to the proximal end of the tibia from multiple points along the joint in the image. The typical measurement points are on the medial and lateral femoral condyles, but the measurement methodology vary on purpose or goal (Beattie, 2008; Keen, 2009; Hayashi, 2016). Particularly, the angle at which the knee joint rests on patients develops with onset OA and can affect the ultrasound measurements (Nagao, 1998). Manual segmentation is seen as the “gold standard” when measuring JSW and serves as the metric of accuracy in research. There are many limitations with using manual data collection of US images. Deviations in transducer placement can introduce more noise in the US image and vary by patient. These deviations can drastically change the placement of the JSW in the image and can interfere with some segmentation methods. Many assisted methods have arisen to combat these deviations.

2.2.2 Assisted Segmentation

Manual Segmentation has well defined limitations in accuracy and throughput of segmentation (Saba, 2018; Faisal, 2017). Determining an accurate segmentation of the joint space is hindered by noise present in the B-mode US image. The most impactful noise in US knee images is speckle noise. Speckle noise results in grainy images that makes determining edges in images difficult (Benzarti, 2013; Michailovich, 2006). The speckle noise in ultrasound images tends to be non-uniform, making simple filters ineffective in removing the noise generated. This is particularly important in diagnosis of Knee OA, as the width between the edges of the joint space are the ROI. Another artifact that appears in this application of US is shadow artifacts. This occurs around hyperechoic regions of the US and serves to blur the hypoechoic regions into having higher intensities than it would otherwise. This blurs edges in US images and prevents a sharp, definitive boundary around linear bone structures (Barr, 2013), like that on the femur and tibia in knee images. Less impactful but other challenges in ultrasound image segmentation include the limited field of vision and attenuation of deeper structures, both unavoidable limitations of transducer and signal detection methods.

Despite the limitations of US images, methods have been proposed to assist or automate segmentation of the JSW in Knee OA. Most methods that seek to assist segmentation of the JSW use a method of enhancing the US image by denoising and improving the contrast of the ROI. There are many different established methods of accomplishing improved images, including: various histogram equalizations (Hossain, 2014; Hossain, 2015; Amorim, 2018; Kim, 1997; Wang, 1999; Chen, 2003; Sim, 2007; Kim, 2008) (Contrast Limited Adaptive Histogram Equalization, Multipurpose Beta Optimized Recursive Bi-Histogram Equalization, etc.) and speckle reduction methods through diffusion methods (Perona, 1990; Yongjian, 2002; Gilboa, 2004; Yu, 2010). The histogram equalization methods hope to enhance the contrast of the ultrasound image by making hypoechoic structures brighter and hyperechoic structures dimmer. The diffusion methods attempt to remove the nonuniform speckle noise in the image without blurring edges or removing smaller details. After image enhancement, the images can then be used for manual segmentation, allowing a cleaner image for a US tech to segment from. To automate the segmentation, ROI finding techniques must be used.

Three proposed methods for automatically segmenting the ROI in the enhanced ultrasound images include: Random Walker (RW), Watershed, and Graph-Cut (Desai, 2018; Desai, 2019). These methods require a distinction to be made between the foreground and background. In the case of RW, a seed point to start growing out the ROI is required. These methods serve to increase the throughput of ultrasound image segmentation, and reach decent Dice Similarity Coefficients of 0.85, 0.82 and 0.81 respectively. The main limitation of these methods is the requirement of selecting seed points to segment from, preventing a truly automatic segmentation of the ROI.

Chapter 3

Methods

3.1 Overview

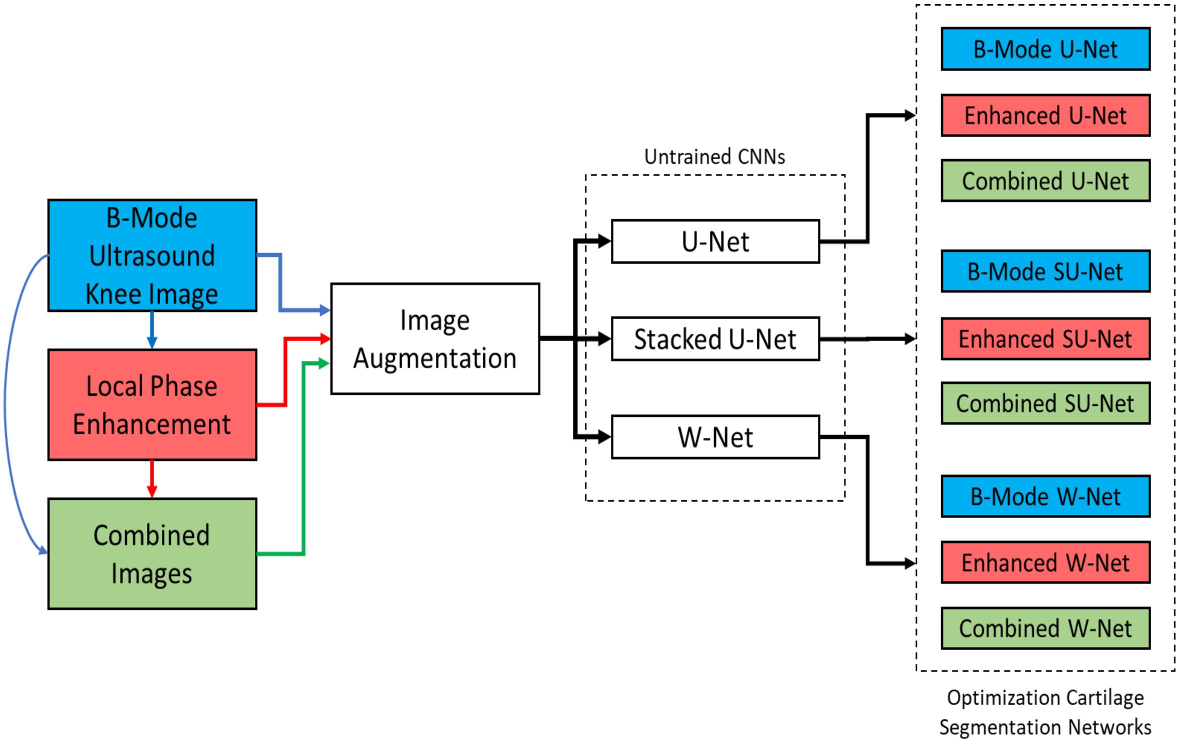


Figure 3.1: A figure showing the workflow of the study. The workflow begins with B-Mode Ultrasound Images and ends with 9 different Neural Networks for every combination of the three possible inputs: B-mode images, Enhanced images and both images combined, along with the three possible networks: U-Net, Stacked U-Net and W-Net.

3.2 Data Collection

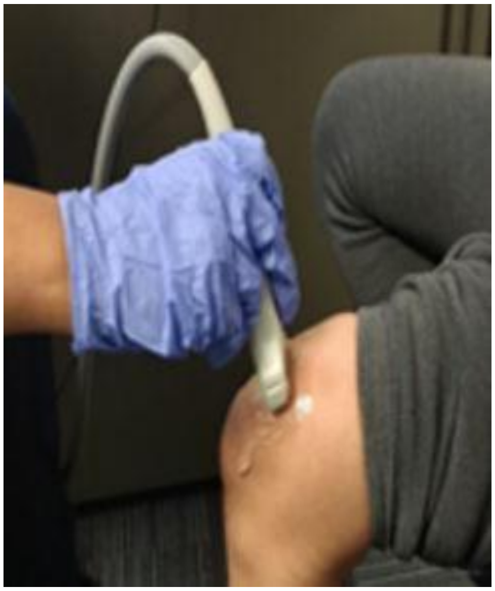


Figure 3.2: A picture showing the method for data collection (Desai, 2018; Desai, 2019). The transducer placement and patient knee position was kept consistent between imaging.

3.2.1 Acquisition

One image dataset used in this study were collected as part of a prior study and repurposed as a comparison between the CNN and the Random Walker (RW) methods (Desai, 2018; Desai, 2019). An example of this dataset, Dataset 1, can be seen in Figure 3.3. The written consent was obtained prior to US scan for total 200 2D images from 10 healthy volunteers. The scans were acquired using 14-5 MHz linear US transducer with a depth setting of 3.5 cm. During the scan, the knee was positioned at 90º of flexion, and US transducer was placed transversely in line with the medial and femoral condyle above the superior edge of the patella. This position of collection can be seen in Figure 3.2. Different scans of cartilage were obtained from both left and right knee joints. Another dataset of 50 images were used for only validation and was obtained using a portable Clarius C3 multipurpose ultrasound transducer using the aforementioned knee positions. An example of images from this dataset, Dataset 2, can be seen in Figure 3.4. The breakdown of the use of the datasets in shown in Table 3.1. These images served to test the performance of the CNNs on an untrained set of data. The training and validation of the Neural Network was implemented in the MATLAB R2018b software package and run on a 4.00 GHz Intel® Core™ i7-8086K CPU, 32 GB 2666 MHz DDR4 RAM PC running Windows 10. Training was performed using the built-in GPU accelerated methods in MATLAB on an NVIDIA GTX 1080.

Table 3.1: A table summarizing the datasets used in this study. There is a total of 200 images in Dataset 1 and 50 images in Dataset 2, making a total of 250 images in this study. Dataset 1 was separated by patient to prevent the same patient being in both the training and validation set.

	Dataset 1	**Dataset 2**
Training	150	0
Validation	50	50
Total	200	50

3.2.2 Cropping

The images from the transducer used in Dataset 1 created images with pixel dimensions of 292 by 380 as seen in Figure 3.3. For the purposes of training the Neural Network, the images were cropped to by 256 by 256. An example of this cropping is seen in Figure 3.5. This cropping was done by simply removing 124 pixel rows away from the hypoechoic values in the bottom of the image and removing 18 pixel columns from each side of the image. The images in Dataset 2 were 800 pixels by 800 pixels. To crop these images, 75 pixels were removed from each side of the image and the image was downscaled to be 256 by 256 using bicubic interpolation and rescaling the pixel values.

The removal of the pixels was necessary to remove the watermarks and depth number from each image.

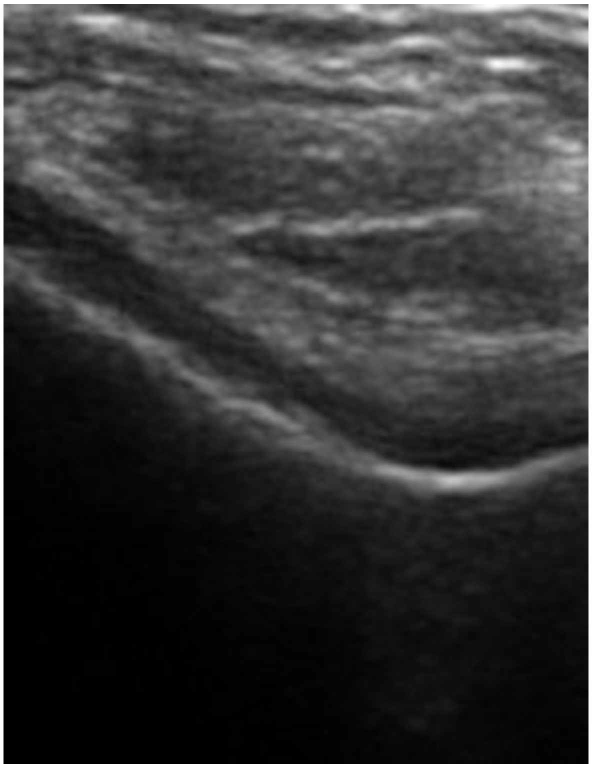


Figure 3.3: An example of the original B-Mode image from Dataset 1.

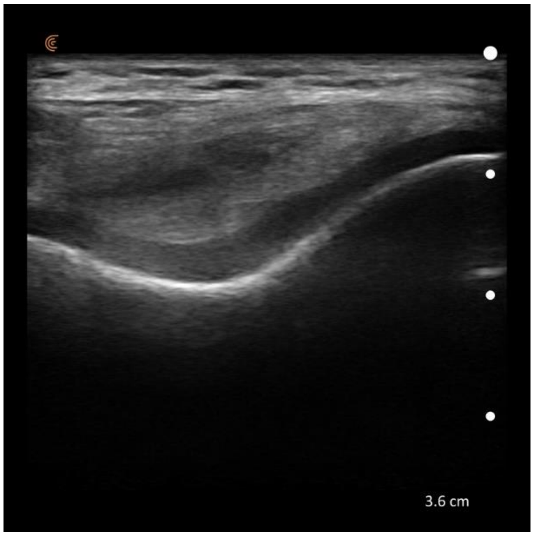


Figure 3.4: An example of the original B-mode image from Dataset 2.

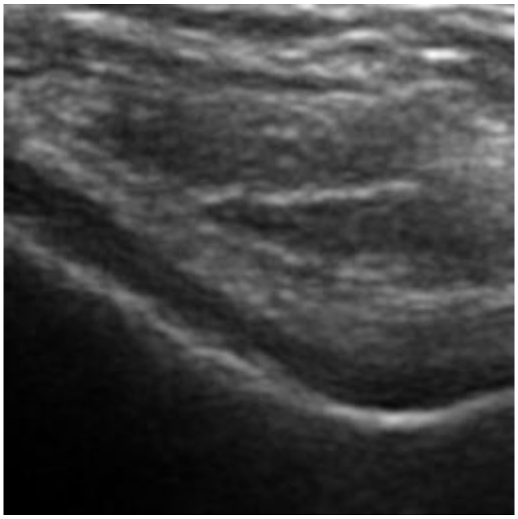


Figure 3.5: An example of cropping the B-mode image in Figure 3.3.

3.2.3 Enhancement

A novel part of this study involves using a previously established method of local phase filtering and bone shadow enhancement on US knee images as an input into the CNNs (Desai, 2019). This method uses a Log-Gabor Filter as a method of edge emphasis of the hyperechoic regions of the knee US image, making the edges of high intensity bone appear more emphasized (Desai, 2019; Boukerroui, 2004). The definition of the Log-Gabor Filter is given in Equation 1.

$$G(\omega, \phi) = \exp\left(-\frac{log\left(\frac{\omega}{\omega_0}\right)^2}{2 * log\left(\frac{k}{\omega_0}\right)^2} + \frac{(\phi - \phi_0)^2}{2 * \sigma_\phi} \right)$$

Equation 1: The polar 2D Log-Gabor Filter as a function of the frequency ω and phase ϕ. Constants ω_0 and ϕ_0 are the center frequency and phase respectively, with width k for the frequency and σ_ϕ for the phase (Desai, 2019).

In order to apply this filter on the US images and output an enhanced image, the discrete Log-Gabor was applied to the (x,y) pixel values of the original B-mode US image to produce the enhanced image, USE(x,y), shown in Equation 2.

$$USE(x,y) = \frac{\sum_r \sum_s \left[[e_{rs}(x,y) - o_{rs}] - T_r \right]}{\sum_r \sum_s \left[\sqrt{\, e_{rs}^2(x,y) - o_{rs}^2(x,y)} \right] + \epsilon}$$

Equation 2: The enhanced image USE(x,y) with the 2D discrete Log-Gabor Filter applied from Equation1 (Desai, 2019). This equation separates the even and odd responses from the Log-Gabor Filter as e(x,y) and o(x,y) respectively. T_r is the noise bias and ϵ is an offset.

After applying this filter to the image in Figure 3.5, the image in Figure 3.6 is produced. This image shows great emphasis on the cartilage boundary, giving sharp boundaries and removing the shadowing effects around the hyperechoic bone structures. Using this image in the assisted RW segmentation proved to significantly increase the performance of segmentation (Desai, 2018). This study seeks to test whether adding these images to the training set and validating with enhanced images would significantly affect the performance of the CNN. This was done by both training CNNs with only the enhanced images as an input into the network and using a combination of both the B-Mode and enhanced images together in the beginning of the CNNs as an early stage fusion.

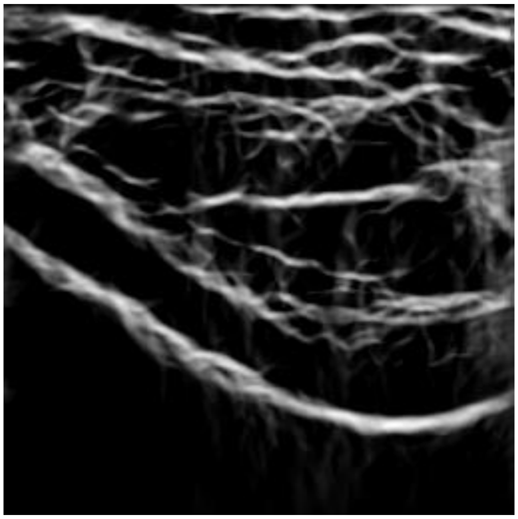


Figure 3.6: The results of using the local phase bone shadow enhancement algorithm on the image from Figure 3.5.

3.2.4 Joint Space Segmentation

The current standard for measuring JSW is to manual segment the ROI from the US image. For the purposes of training and validation of the CNN, each image was segmented using the built in Image Segmenter in Matlab. This allowed a tracing of the JSW over the original US B-mode image and created a binary image shown in Figure 3.7. These images served as the ground truth for the network training and validation after training was complete. The segmentation is not necessarily a perfect measurement of the true cartilage thickness in the patient, as there is natural noise in boundaries that exist in the US image. The only method of measuring the true cartilage thickness is through invasive surgery or using a cadaver. The true JSW measurement can also vary between images in the same patient, leaving variability in manual segmentations. For the purposes of training and evaluating the networks, the manually segmented images will be taken as the ground truth, although it is possible for a segmentation algorithm to perform a more accurate segmentation than a manual one.

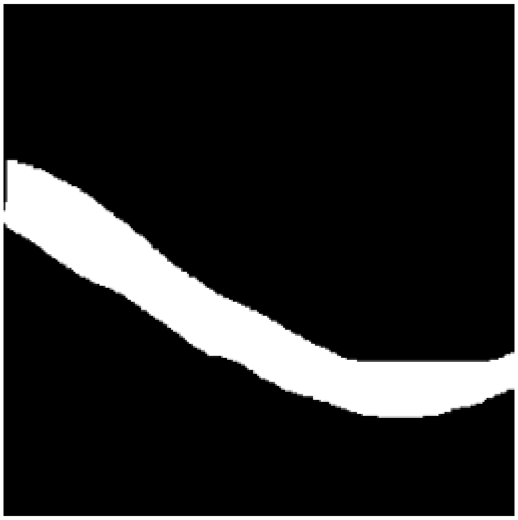


Figure 3.7: The manually segmented knee cartilage of Figure 3.5. This was used for training the neural networks.

3.2.5 Image Augmentation / Data Structure

Training a predictive CNN can require many unique samples to train in order to reach an accurate result. The datasets used in this study contained 200 and 50 US images in total. This is a small database to train a large-scale CNN (Russel, 1995). In order to increase the total number of images for training, a method of modifying the images was developed. Firstly, a data structure for saving the original dataset images was modified starting with the Matlab Datastore class. This class was modified to allow random drawing of images when prompted. These images were then given a 25% chance of independently being augmented in three ways: translation, rotation or mirroring. The first modification was to translate or slide the image using linear interpolation. The image was translated both horizontally and vertically a random amount, with a maximum translation of 10 pixels in either direction. The second modification was to rotate the image using nearest-neighbor interpolation. The bounds of rotation were 10 degrees either clockwise or counterclockwise. The final modification was to flip the image horizontally. The choices of maximum modifications attempted to keep the similarity of the collected images to images that would be reasonable to collect in a real US image. These modifications gave a total possible number of unique training images of 120,000. These modifications served to not only increase the total number of possible images in the dataset, but also increased the robustness of the CNN and prevented overfitting from making the CNN recognize images in the dataset rather than features in those images. These modifications were applied to both images that were being inputted into the CNN and the ground truth image. Although Matlab has these properties implemented within the built-in library, categorical ground truth images used for training were incompatible and were found to increase training time.

3.3 Neural Network Architectures

There are many well established CNNs that have been used for medical image segmentation. Many of these are applied to other modalities such as MR or Computed Tomography (CT). Many of the medical imaging CNNs for segmentation use the U-Net architecture (Ronnesberger, 2015). This architecture has been improved to fit different applications and has given rise to both the Stacked U-Net and W-Net as a result. These networks generally have more learnables, weights and biases, that theoretically improve segmentation power of the CNN. These networks were chosen for this study, however, because of the crosstalk across the network architectures that allow for connections that skip over the deep layers of the network. This crosstalk should accentuate the local phase enhancement of the input images to give a more direct effect on the final segmentation.

3.3.1 U-Net

The U-Net used in this study uses various feedforward convolutions to take an input of 256x256xN, where N is the number of images fed into the U-Net, to create a 256x256x2 probability distribution matrix of whether a pixel is part of the ROI or the background. The U-Net architecture can be seen in Figure 3.8. The U-Net was implemented in Matlab using the various layer commands. All the learnables, weights and biases, were randomized prior to training. The U-Net goes through 4 different layers of feature sizes to capture features of different sizes throughout the image (Ronnesberger, 2015). These important features are copied over in the earlier parts of the network and propagated forward towards the end of the network. The sizes chosen for each matrix were made to be halves of each previous layer and the number of downsampling steps were made to four to not overload the GPU memory during training.

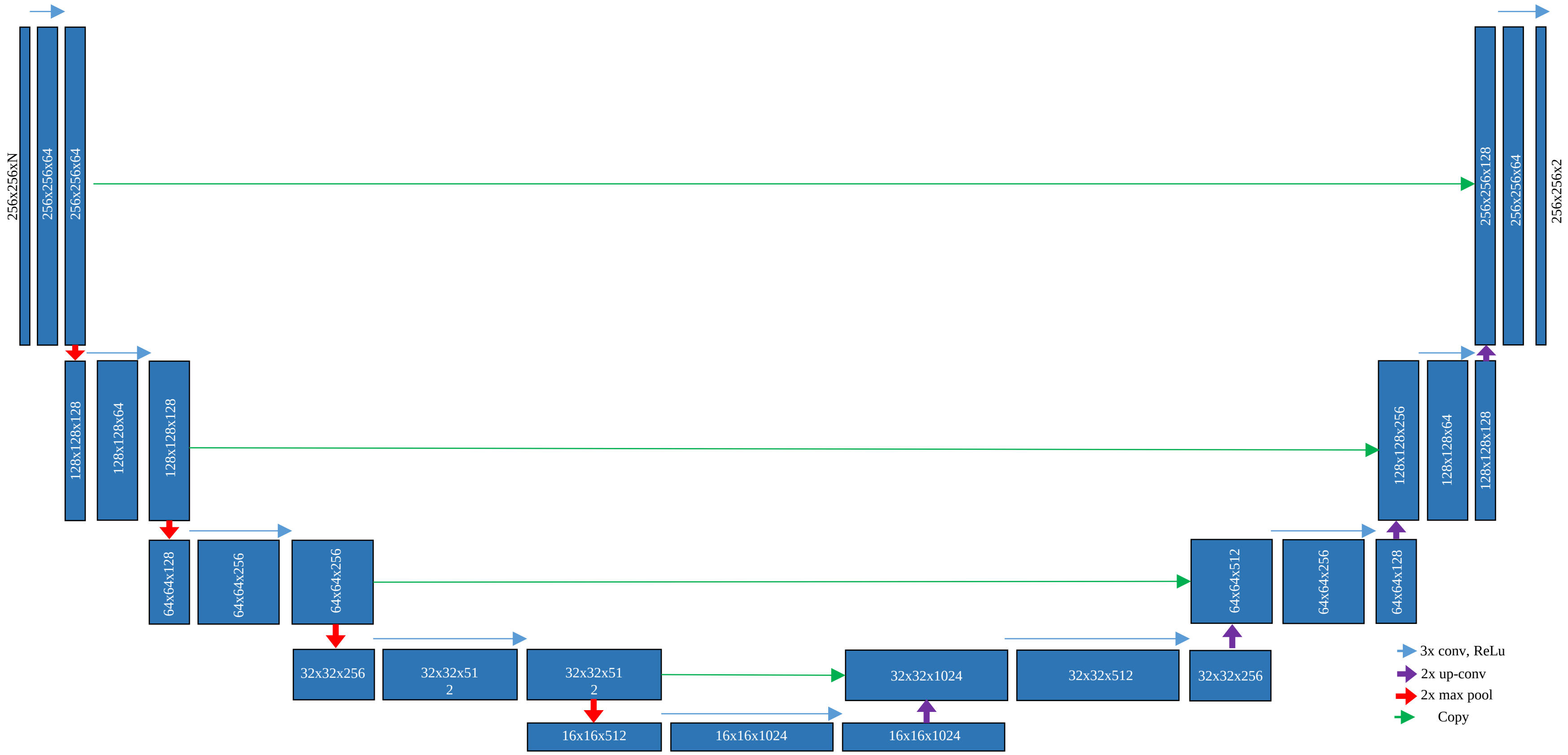


Figure 3.8: The network architecture of the U-Net used in this study. The input into the U-Net has a size of 256x256xN, where N is the number of input images. If only the B-mode image is being sent into the U-Net, then it is 256x256x1. This image tile is then propagated through the net through a series of two 3x3 convolutions (3x conv) each followed by a Rectified Linear Unit (ReLu) which simply sets any negative value in the matrix to zero. This matrix is then downsampled using a 2x2 max pooling (2x max pool) which uses a 2x2 mask and returns the maximum value in the mask to reconstruct the matrix at approximately half the size. This process is then repeated until a size of 16x16x1024, where the matrix is then upsampled by using a 2x2 up convolution (2x up-conv) and concatenating it with a copy of the previous 32x32x512 matrix. This process continues until the final layer of 256x256x64 which is convolved with a 1x1x64 matrix to make a final size of 256x256x2. This final matrix is then softmaxed to create another 256x256x2 matrix which serves as the probability distribution of a given pixel in the original image being the ROI. The network has 58 total layers.

3.3.2 Stacked U-Net

Since the original publication of the U-Net in 2015 as a method of medical segmentation, there have been various changes and manipulations to the U-Net architecture (Wang, 2018; Tang, 2018; Sun, 2018; Shah, 2018; Milletari, 2016). Most of these changes focused on adding more layers or more dense connections between nodes in the network. One implementation that showed to be promising was the Stacked U-Net or SU-Net (Shah, 2018). This implementation stacks the output of a U-net (Ronnesberger, 2015) and feeds it into another U-Net. A similar network to this was implemented in this study and can be seen in Figure 3.9. The original work intended for multiple U-Nets to be chained together, up to sixteen in the original publication. For this implementation with the limitation of GPU memory, only two U-Nets were bridged. In order to bridge the U-Nets, the output of the original U-Net in Figure 3.8 that was a 256x256x64 matrix was bridged with the input of a second U-Net with a matrix of 256x256x64. This connection was a 3x3 convolution followed by a ReLu. Comparing the SU-Net to the original will show if connecting the additional U-Net would supply any significant improvements to the predictive power of the CNN.

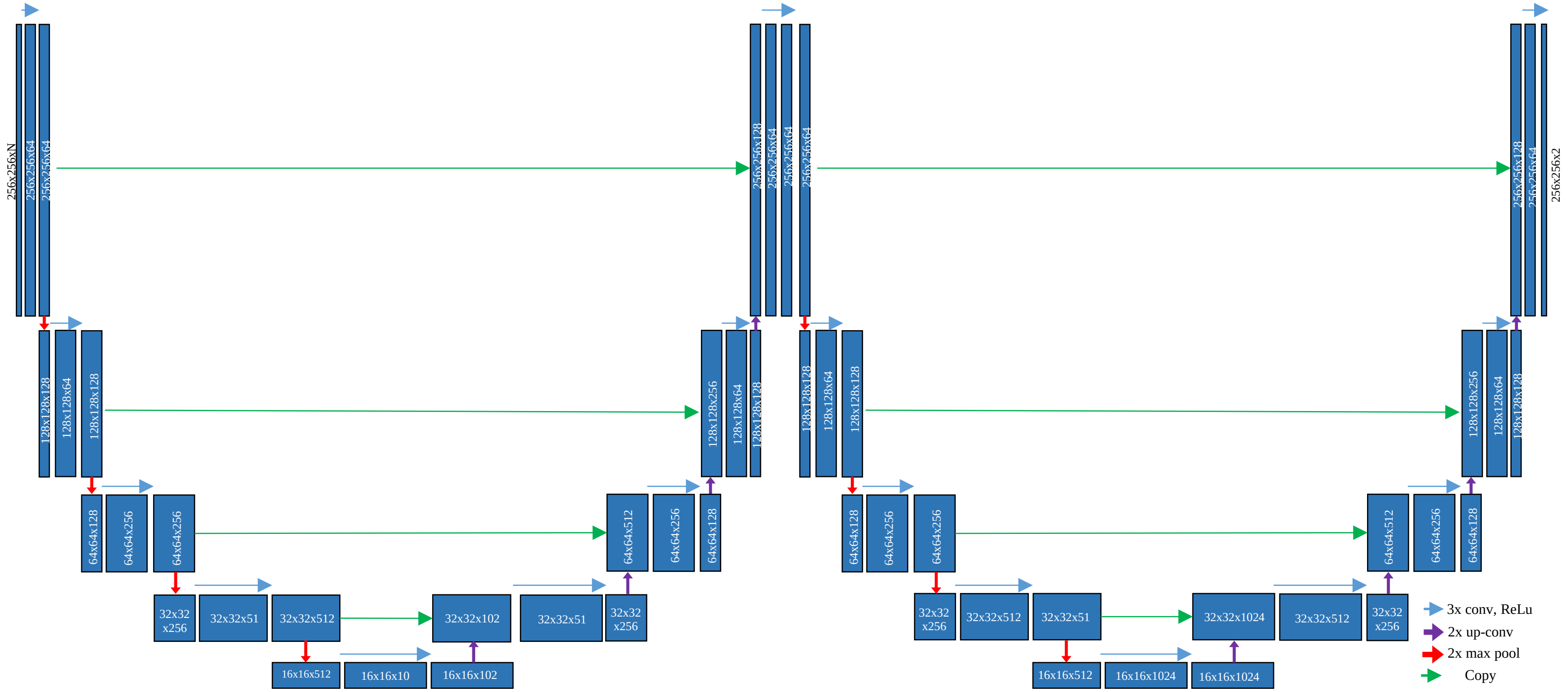


Figure 3.9: The network architecture of the SUNet used in this study. This architecture mostly follows the architecture of Figure 3.8, with a bridge in the center of the path that uses a 3x3 convolution followed by a Rectified Linear Unit to bridge the 256x256x64 matrices together. The input of the network is like the U-Net and accepts an input of 256x256xN where N is the number of images. The network outputs a 256x256x2 probability distribution image of whether the pixel is in the foreground or background. The network had 110 layers in total.

3.3.3 W-Net

A final network to be implemented in this study is a recent expansion on the idea of SU-Nets. This network takes two bridged U-Nets and bridges them along the copy connections across the gap between the networks (Xia, 2017; Chen, 2019). This new net is called either “bridged U-Nets” or “W-Net.” To prevent confusion with other CNNs in this study, it will be referred to as the W-Net in this study. This network was used for 2D prostate segmentation in MR images and showed a significant increase in DSC than a similarly weighted SU-Net. This network was implemented and can be seen in Figure 3.10. The goal of this net was to emphasize the features that were being copied across the first U-Net closer to the output of the total network to not lose the data through the network. This addition over the U-Net would also decrease training time and increase the robustness of the network. The addition of the addition block did, however, increase the total size of the network in memory. Comparing this network to the SU-Net will show how important preserving the copying blocks is to the segmentation power of the network.

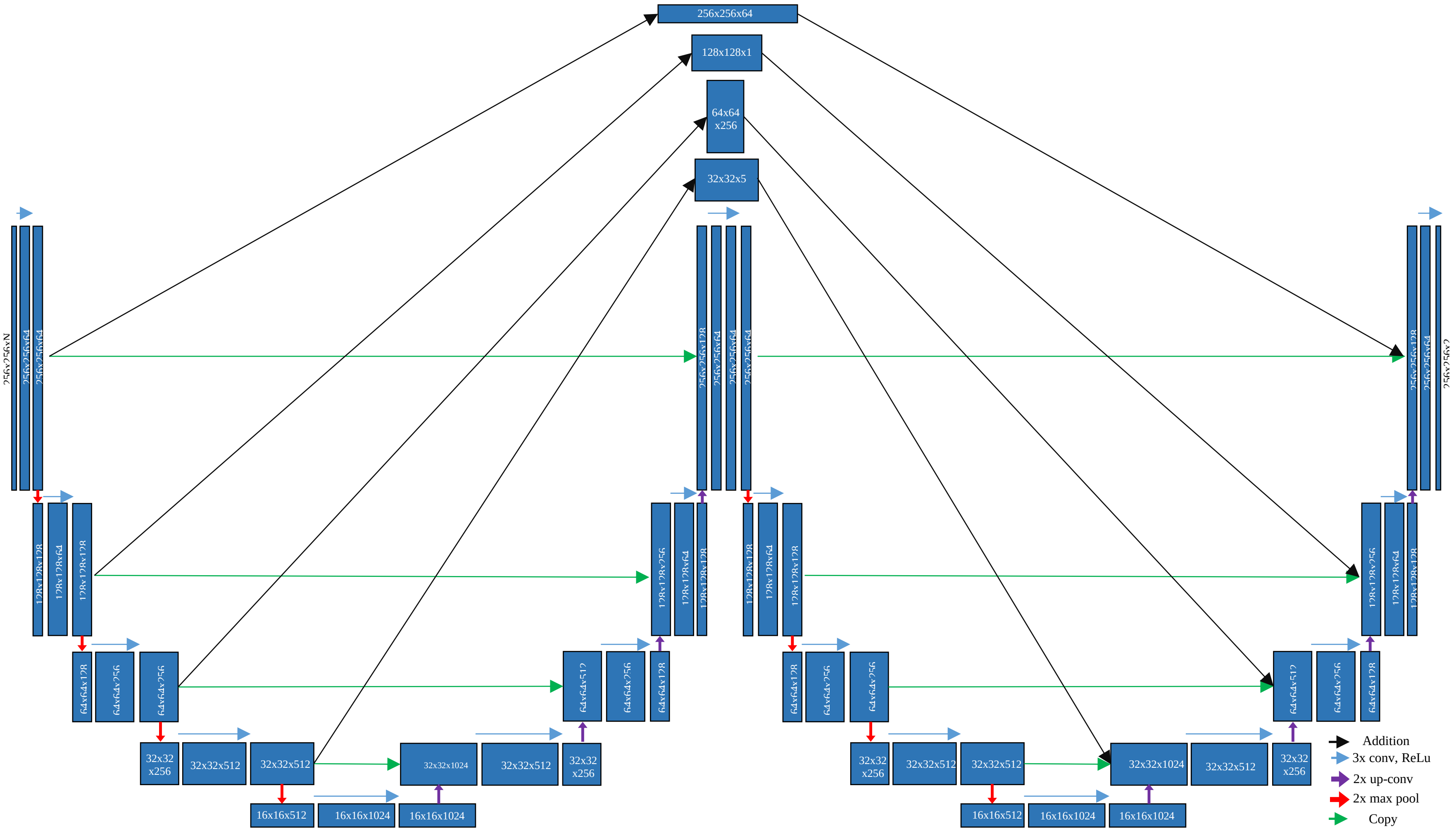


Figure 3.10: The network architecture of the W-Net used in this study. This network is a slightly modified version of the SUNet from Figure 3.9. The matrices that are copied within in the first U-Net are now also stored in another layer and then added to the matrix of similar size in the second U-Net prior to the up-convolution. This serves to amplify features found in the first U-Net while still preserving the deeper features captured in the second U-Net. The input of the network is like the U-Net and accepts an input of 256x256xN where N is the number of images. The network outputs a 256x256x2 probability distribution image of whether the pixel is in the foreground or background. The network had 114 layers in total.

3.3.4 Training Methods

To train the CNNs in this study, 150 of the 200 images from Dataset 1 were separated and used as a training set. The CNNs in this study were trained using the Matlab GPU accelerated training algorithms. The parameters of updating the weights and biases were kept consistent throughout the training of all networks to give more insight into the comparison of how the networks trained. Two different methods were initially considered to be used: Stochastic Gradient Descent with Momentum (SGDM) and Adam (Russel, 1995; Kingma, 2014). The main difference between these two methods is that SGDM uses a single learning rate, or weight per iteration, on the entire backpropagation of error for all parameters, while Adam uses an adaptive model that changes the learning rate for each parameter (Kingma, 2014). Although Adam can approach a similar result to SGDM it has potentially less training iterations. The initial learning rate, however, was found to produce inconsistent results depending on the network architecture being trained. If an initial learning rate was found to be successful in a U-Net, it caused unsuccessful training in the W-Net. Since the parameters were kept consistent to allow for equal comparisons, SGDM was used instead. The update algorithm for SGDM is defined in Equation 3.

$$\theta_{i+1} = \theta_i - \alpha\nabla\mathrm{E}(\theta_\mathrm{i}) + \gamma * (\theta_i - \theta_{i-1})$$

Equation 3: The definition of Stochastic Gradient Descent with Momentum. θ_x represents the x iteration of parameter vector (weight or bias) that is being updated, α represents the current learning rate for the training, $\nabla\mathrm{E}(\theta_i)$ represents the gradient of the loss function or the total rate of error accumulation in a training epoch, and γ represents the momentum factor.

SGDM was implemented using a custom mini-batching system mentioned in Section 3.1.5. This augmenting system randomly picked 10 images from the dataset, augmented the images, and calculated the loss function across the 10 images in a minibatch. This minibatch was used to calculate the loss function and dynamically update the weights and biases in the network. The parameters used for training was an initial learning rate, α, of 0.05, a momentum factor, γ, of 0.9, a regularization of loss factor (L_2 Regularization) of 0.0001 to help prevent gradient explosion, and a maximum number of training epochs of 20. Many network trainings did not reach the maximum of 20 epochs and this was implemented to prevent overfitting in poorly training networks. The average number of epochs was 16 across all networks. Two different training curves are shown in Figure 3.11 and represent both abnormally successful and unsuccessful network training. The unsuccessful training was ended before training could be completed because of gradient explosion, an issue that can arise with some deep-learning trainings (Yang, 2018). This occurs rarely when reaching large loss values that lead to overflows in the SGDM calculation. Matlab has a method of detecting this event and stopping training from continuing. Gradient explosion was a rare event and only mostly occurred when choosing extreme training parameters for learning rate and loss factors. Like abnormally unsuccessful trainings, some trainings were abnormally successful as shown in the bottom graph of Figure 3.11. This training curve plateaued very quickly since the network started with an 80% accuracy and produced the best performing network in this study. A typical training curve can be seen in Figure 3.12. This training graph shows an initially unsuccessful training result with an initial accuracy of 8.8%, but a promising 90% accuracy by the first 100 iterations. Despite having a high classification accuracy, the Dice Similarity Coefficient for these segmentations would be relatively low at this stage. The network at 100 iterations has established the location of the knee cartilage but has not learned to segment the finer details and boundaries at this stage of training. As seen in the successful training in Figure 3.12, the training accuracy tends to oscillate around 90% accuracy, but this accuracy can be misleading in terms of finer details of segmentation. This would affect the quality of the segmented region and can lead to noise within segmented shapes and noisy boundaries. The network is learning finer details that cannot be shown using a simple accuracy measurement, but only through validation.

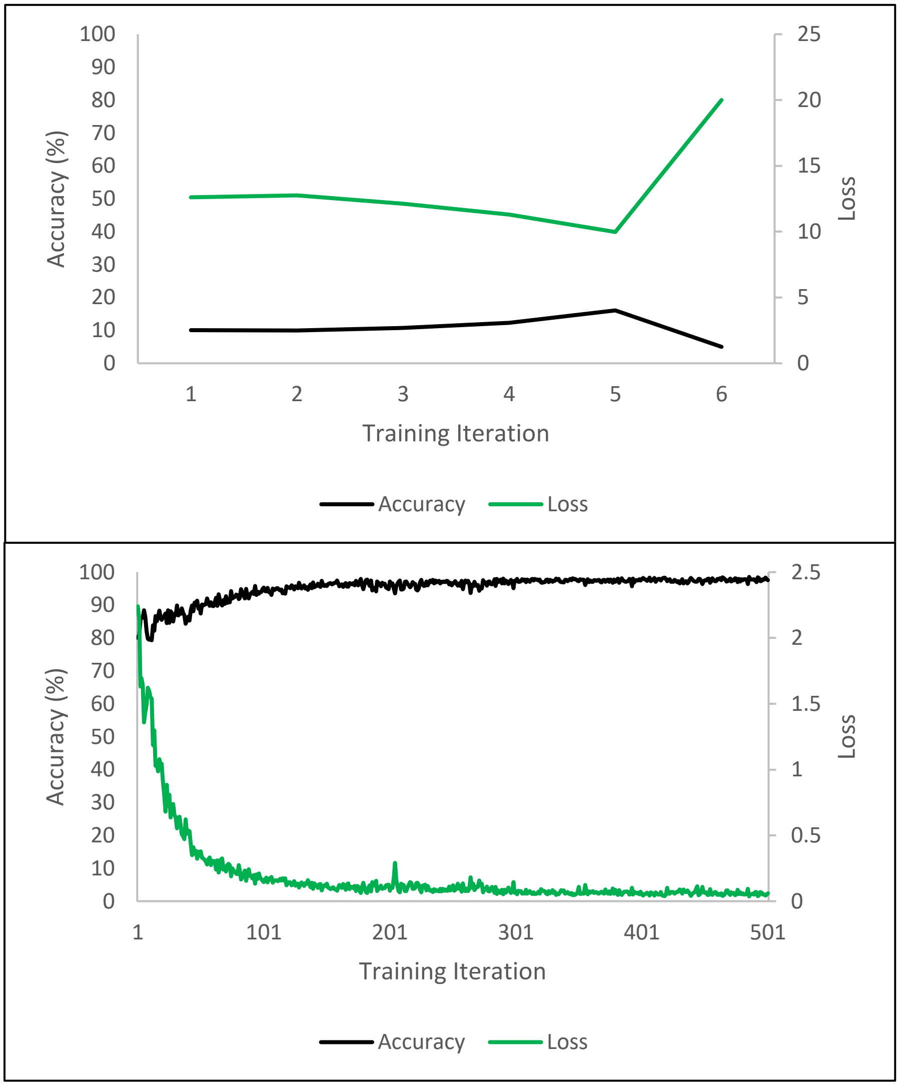


Figure 3.11: A graph of the training information from a failed Stacked U-Net training (top) and a successful W-Net training (bottom). The training accuracy in black is calculated as the percentage of pixels that were correctly classified into the correct group of “foreground” or “background”. The Loss in gray is calculated as the cross-entropy loss in the prediction. The top failed graph shows a training sequence that led to gradient explosion and premature finalization of training for the network before reaching a meaningful result. The bottom graph shows the first 500 iterations of a training set that started with an abnormally high accuracy of 80.%.

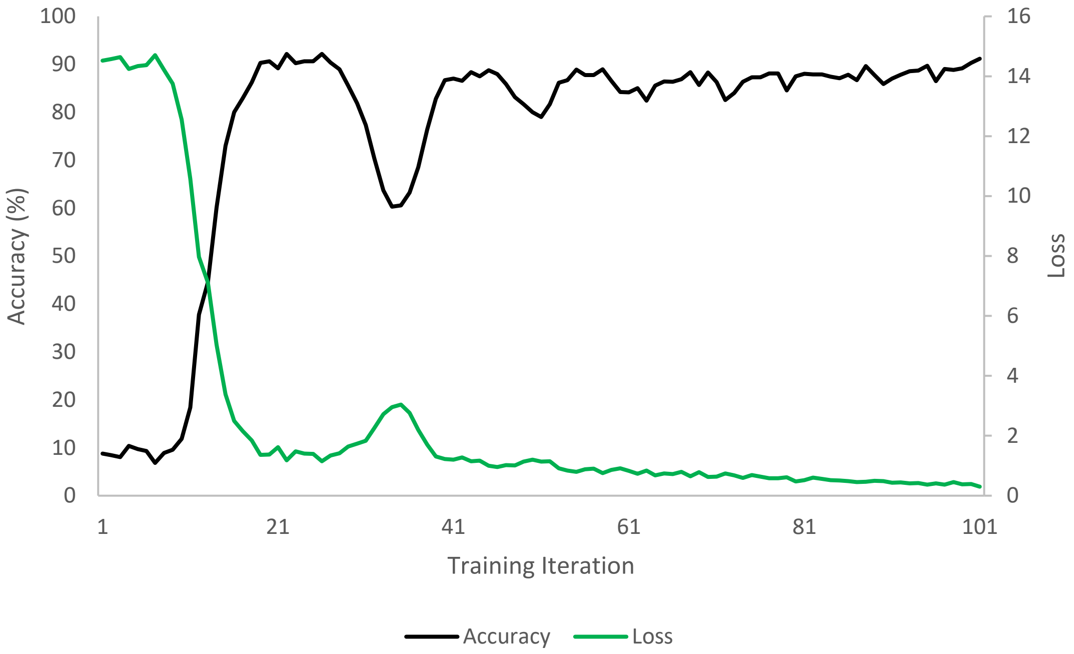


Figure 3.12: A training graph showing the typical beginning of a training curve. This network was training a U-Net with only B-mode ultrasound images. The accuracy is shown in black and starts at a low accuracy of 8.8% and increases to 90.2% by the first 100 iterations of training.

Network training for this study had produced inconsistent results in some networks. To prevent abnormal results from affecting conclusions to be drawn from problems such as gradient explosion and abnormal starting network parameters, a 10-fold cross validation was done on every combination of network. Each combination of network and input type was tested by training 10 different networks with randomized starting parameters, training image orders and augmentations on the training images. After training, these networks were validated using two different datasets: the remaining 50 images from Dataset 1 using similar images to training, and the 50 images from Dataset 2, a separate image set with a different transducer and system. The file sizes of the networks were: 113 MB for a U-Net, 225 MB for a SU-Net and 226 MB for a W-Net on average after training.

3.4 Analysis Metrics

In order to test the accuracy of the trained Neural Networks, the manually segmented images were used as the ground truth, and a Dice Similarity Coefficient (DSC) was used to compare the results of the prediction. The DSC is used as a way of measuring how similar two binary images are. The DSC is defined in Equation 4 (Dice, 1945).

$$Dice(A,B) = 2\frac{|A \cap B|}{|A| + |B|}$$

Equation 4: The Dice Similarity Coefficient of predicted image A and ground truth image B. This calculates the amount of pixel overlap there is between A and B as a ratio by how many total pixels there are in both images, and scaled by how many images are being compared to normalize, 2. The DSC can range from 0 to 1, where 0 is no similarity and 1 is the same image.

The DSC is a widely adopted measure of semantic segmentation accuracy. For calculations in this study, the image generated from the network, A, was compared against the same image from the manual segmented joint space segmentations from Section 3.1.4, B. A Jaccard coefficient was also calculated for every network but was excluded because of the same trends that were shown with both metrics and would be redundant. The Neural Networks return an output matrix that corresponds to a probability that a position in the matrix belongs to the foreground. In this study, the networks return a 256 by 256 matrix that corresponds to the pixels from the 256 by 256 input image. In order to calculate the DSC, this probability is thresholded to produce a binary image. The threshold value used is 0.5. Any pixel that the network predicts with a 50% or greater confidence is in the foreground is given a high value of 1. Any other pixel in the image was given a low value of 0.

Chapter 4

Results

4.1 Dice Coefficients

The DSC generated from the 90 trained networks on both validation datasets are shown in Tables 4.1, 4.2, and 4.3. The average DSC and standard deviation for every combination of network input and network architecture are shown for the dataset of images similar to those used to train the networks (Dataset #1) and the dataset of images not used to train the network (Dataset #2).

Table 4.1: A table showing the DSCs from the trained U-Nets in this study rounded to two significant digits. Each combination of network was trained 10 times, displayed as a Net Index. The DSCs are divided by the network input types: The raw B-mode images, the Enhanced images, and the combination of both latter. The DSC are then further divided into results from the Dataset of images like those used to train the network, Dataset #1, and the Dataset of images using a different transducer, Dataset #2. Below these combinations is the average of the Dice Coefficients of the 10-fold cross validation, along with the standard deviation.

U-Net						
	B-Mode		Enhanced		Combined	
Net Index	Dataset # 1	Dataset # 2	Dataset # 1	Dataset # 2	Dataset # 1	Dataset # 2
1	0.88	0.83	0.88	0.85	0.89	0.85
2	0.73	0.64	0.63	0.51	0.89	0.87
3	0.87	0.83	0.88	0.78	0.89	0.85
4	0.87	0.78	0.86	0.72	0.87	0.78
5	0.83	0.70	0.87	0.81	0.86	0.77
6	0.87	0.78	0.89	0.82	0.88	0.82
7	0.85	0.76	0.85	0.72	0.87	0.75
8	0.81	0.71	0.88	0.78	0.89	0.83
9	0.84	0.74	0.90	0.84	0.89	0.84
10	0.87	0.78	0.81	0.78	0.88	0.83
Average	**0.841±0.04**	**0.756±0.06**	**0.846±0.074**	**0.762±0.09**	**0.882±0.01**	**0.818±0.04**

Table 4.2: A table showing the DSCs from the trained Stacked U-Nets in this study rounded to two significant digits. Each combination of network was trained 10 times, displayed as a Net Index. The DSCs are divided by the network input types: The raw B-mode images, the Enhanced images, and the combination of both latter. The DSC are then further divided into results from the Dataset of images like those used to train the network, Dataset #1, and the Dataset of images using a different transducer, Dataset #2. Below these combinations is the average of the Dice Coefficients of the 10-fold cross validation, along with the standard deviation. The 1st network in the Enhanced inputs performed the worst in the entire study and was the result of a gradient explosion during training as mentioned in Section 3.2.4. The 8th network in the combined inputs is tied for the highest DSC in the study. Excluding this outlier gives an average DSC of 0.890±0.02 for the first dataset and 0.811±0.05 for the second dataset across the 9 networks.

Stacked U-Net						
	B-Mode		Enhanced		Combined	
Net Index	Dataset # 1	Dataset # 2	Dataset # 1	Dataset # 2	Dataset # 1	Dataset # 2
1	0.60	0.48	0.19	0.15	0.87	0.72
2	0.87	0.74	0.89	0.81	0.89	0.79
3	0.84	0.63	0.87	0.73	0.90	0.76
4	0.78	0.56	0.90	0.88	0.90	0.85
5	0.78	0.60	0.90	0.79	0.91	0.83
6	0.78	0.65	0.91	0.85	0.84	0.75
7	0.67	0.55	0.89	0.79	0.91	0.83
8	0.67	0.53	0.91	0.85	0.91	0.84
9	0.89	0.82	0.91	0.86	0.91	0.82
10	0.84	0.62	0.84	0.74	0.91	0.73
Average	**0.771±0.09**	**0.619±0.10**	**0.820±0.2**	**0.745±0.2**	**0.896±0.02**	**0.792±0.05**

Table 4.3: A table showing the DSCs from the trained Stacked U-Nets in this study rounded to two significant digits. Each combination of network was trained 10 times, displayed as a Net Index. The DSCs are divided by the network input types: The raw B-mode images, the Enhanced images, and the combination of both latter. The DSC are then further divided into results from the Dataset of images like those used to train the network, Dataset #1, and the Dataset of images using a different transducer, Dataset #2. Below these combinations is the average of the Dice Coefficients of the 10-fold cross validation, along with the standard deviation. The 2nd network in the combined dataset is tied for the highest DSC in the entire study.

W-Net						
	B-Mode		Enhanced		Combined	
Net Index	Dataset # 1	Dataset # 2	Dataset # 1	Dataset # 2	Dataset # 1	Dataset # 2
1	0.81	0.70	0.90	0.85	0.91	0.82
2	0.83	0.68	0.90	0.84	0.91	0.84
3	0.81	0.67	0.90	0.81	0.86	0.73
4	0.81	0.72	0.89	0.77	0.91	0.84
5	0.79	0.67	0.90	0.84	0.87	0.77
6	0.77	0.66	0.91	0.84	0.89	0.81
7	0.85	0.67	0.90	0.78	0.88	0.79
8	0.88	0.77	0.91	0.84	0.90	0.82
9	0.72	0.58	0.89	0.79	0.91	0.83
10	0.83	0.75	0.90	0.83	0.91	0.83
Average	**0.809±0.04**	**0.686±0.05**	**0.900±0.007**	**0.820±0.03**	**0.895±0.02**	**0.810±0.03**

4.2 Grouping by Input

To better visualize the impact of the choice of input into the networks, Figure 4.1 shows the mean DSCs for Dataset 1, and Figure 4.2 shows the mean DSCs for Dataset 2. The most consistent networks trained resulted from the networks that took a combination of the B-mode image and the enhanced image as an input across both datasets. The highest average DSC across all combinations, however, resulted from using the enhanced images in a W-Net in the second dataset. Figures 4.3 and 4.5 show the best segmentations from the different inputs for Dataset 1 and Dataset 2 respectively. These segmentations show promising results, with the Dataset 1 results showing very smooth boundaries around the segmentation in the enhanced and combination image input networks. Figure 4.4 shows the segmentations generated from the lowest DSC networks for Dataset 1. This includes the one SU-Net that had a gradient explosion in training. The B-mode inputs showed very messy segmentation and many false positives outside the desired ROI in this image. The combination network shows decent results, with one artifact in the original image showing as a false positive. These Figures also included segmentations from the RW algorithm that Dataset 1 was originally used in (Desai, 2018). In Figure 4.5, however, the RW parameters were not set optimally for Dataset 2 to highlight that it would need to be tuned to perform as well as Dataset 1.

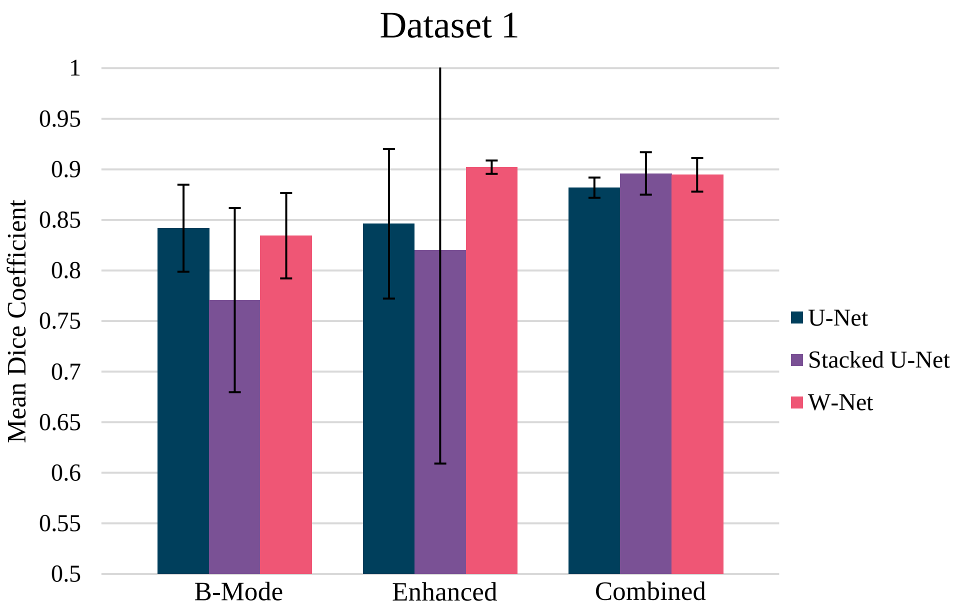


Figure 4.1: A graph showing the mean Dice Similarity Coefficient validated on Dataset #1 for the 10 networks in each combination sorted by the input into the network. Error bars show the standard deviation of the DSC across each set of 10 networks. One outlier in the Enhanced Stacked U-Net produced a large standard deviation and notably lower DSC for that set of networks. The highest average and lowest standard deviation was achieved by the combined images input networks.

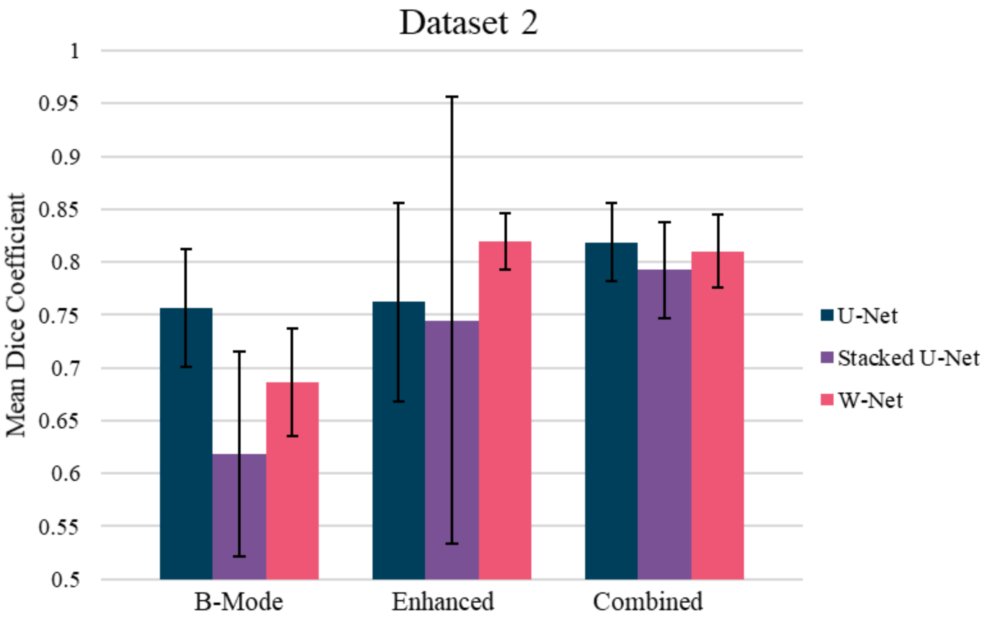


Figure 4.2: A graph showing the mean Dice Similarity Coefficient validated on Dataset #2 for the 10 networks in each combination sorted by the input into the network. Error bars show the standard deviation of the DSC across each set of 10 networks. One outlier in the Enhanced Stacked U-Net produced a large standard deviation and notably lower DSC for that set of networks. The highest average and lowest standard deviation across inputs was achieved by the combined images input networks, the highest single combination was the W-Net using the Enhanced images as an input only.

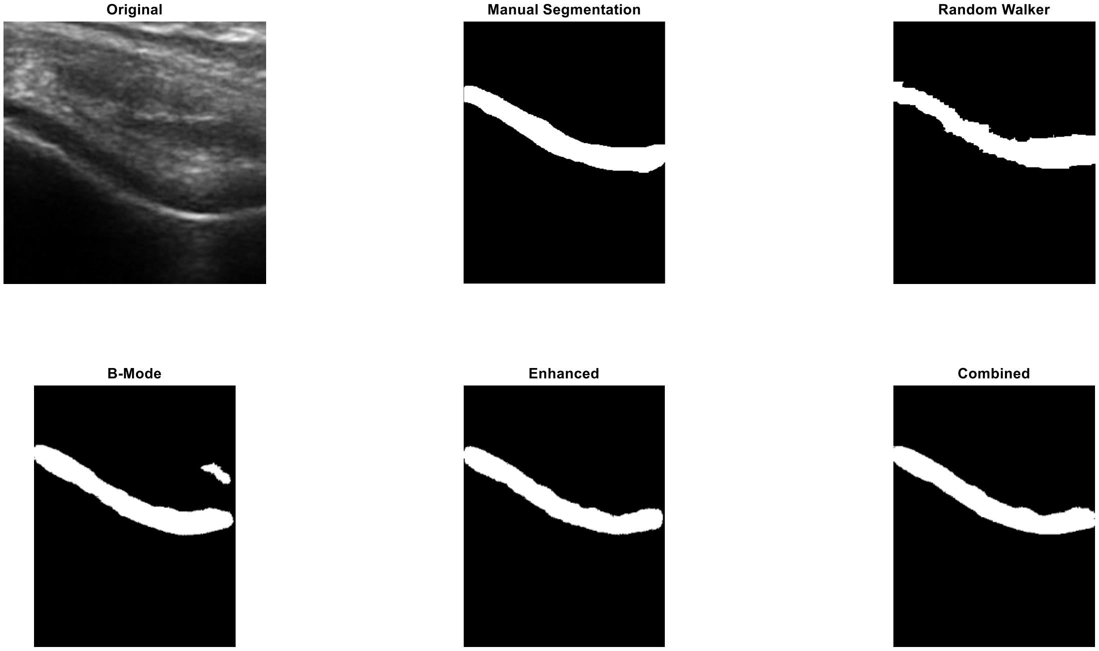


Figure 4.3: Example segmentations from dataset 1 from the best networks across the three different inputs. The three lower segmentations were generated from a Stacked U-Net using a B-mode image input, an Enhanced image input and a combined image input. The manual segmentation was created using the Image Segmenter in Matlab, and the Random Walker segmentation was generated using an algorithm used on this dataset previously (Desai, 2018). The Random Walker shows a blocky segmentation while the Network segmentations show smoother boundaries. The Dice coefficients for the B-mode, Enhanced and Combined images are: 0.8828, 0.9180 and 0.9265 respectively.

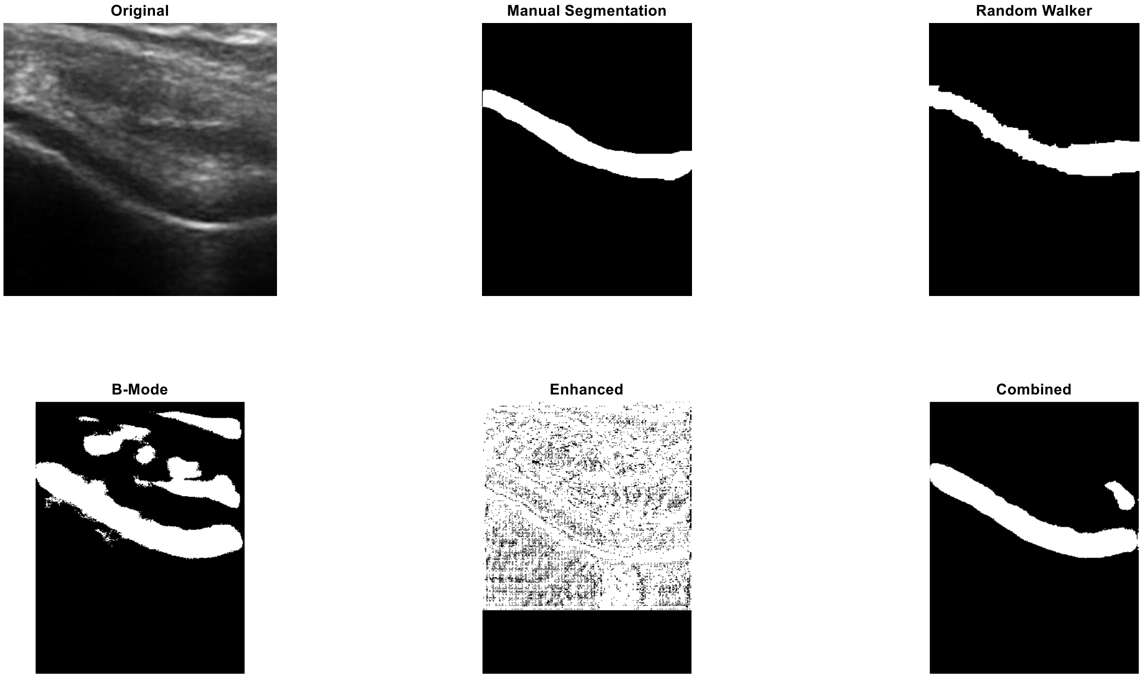


Figure 4.4: Example segmentations from dataset 1 from the worst networks across the three different inputs. The B-Mode and Enhanced image segmentations were generated using a Stacked U-Net. The Combined segmentation was generated using a W-Net. The enhanced segmentation used the network that terminated early during training because of gradient explosion, resulting in noise. The manual segmentation was created using the Image Segmenter in Matlab, and the Random Walker segmentation was generated using an algorithm used on this dataset previously (Desai, 2018). The Random Walker shows a blocky segmentation while the Network segmentations show smoother boundaries. The Dice coefficients for the B-mode, Enhanced and Combined images are: 0.6099, 0.2042 and 0.8819 respectively.

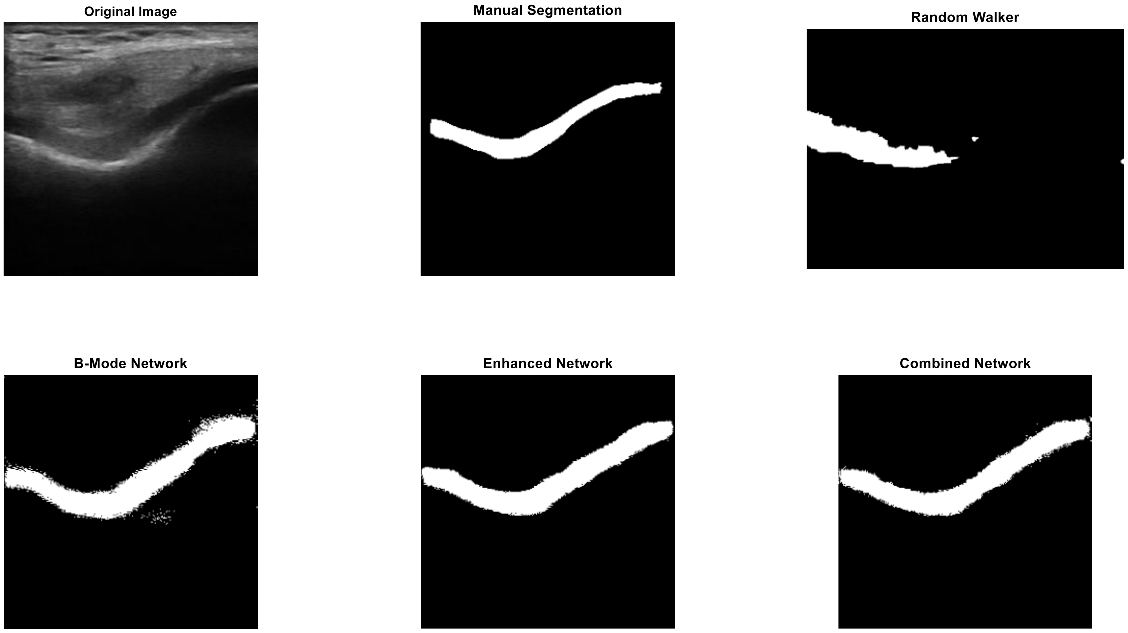


Figure 4.5: Example segmentations from dataset 2 from the best networks across the three different inputs. The B-Mode and Combined image segmentations were generated using a U-Net. The Enhanced segmentation was generated using a Stacked U-Net. The manual segmentation was created using the Image Segmenter in Matlab. The Random Walker segmentation in mostly unusable as the algorithm was not created for this dataset and needs different parameters to perform optimally. The Dice coefficients for the B-mode, Enhanced and Combined images are: 0.6668, 0.7103 and 0.7188 respectively.

4.3. Grouping by Network

To better visualize the impact the network architectures had on the final network, Figure 4.6 shows the mean DSCs for Dataset 1, and Figure 4.7 shows the mean DSCs for Dataset 2. The most consistent networks trained resulted from the networks that took a combination of the B-mode image and the enhanced image as an input across both datasets. The highest average DSC across all combinations, however, resulted from using the enhanced images in a W-Net in the second dataset. Figures 4.8 and 4.9 show the best segmentations from the different inputs for Dataset 1 and Dataset 2 respectively across the different network architectures. All segmentations in these images look very similar between network architectures. The major differences being that the U-Net segmentations tend to have a less smooth boundary on the segmentation and false positives in both the B-Mode U-Net and the Enhanced Image W-Net.

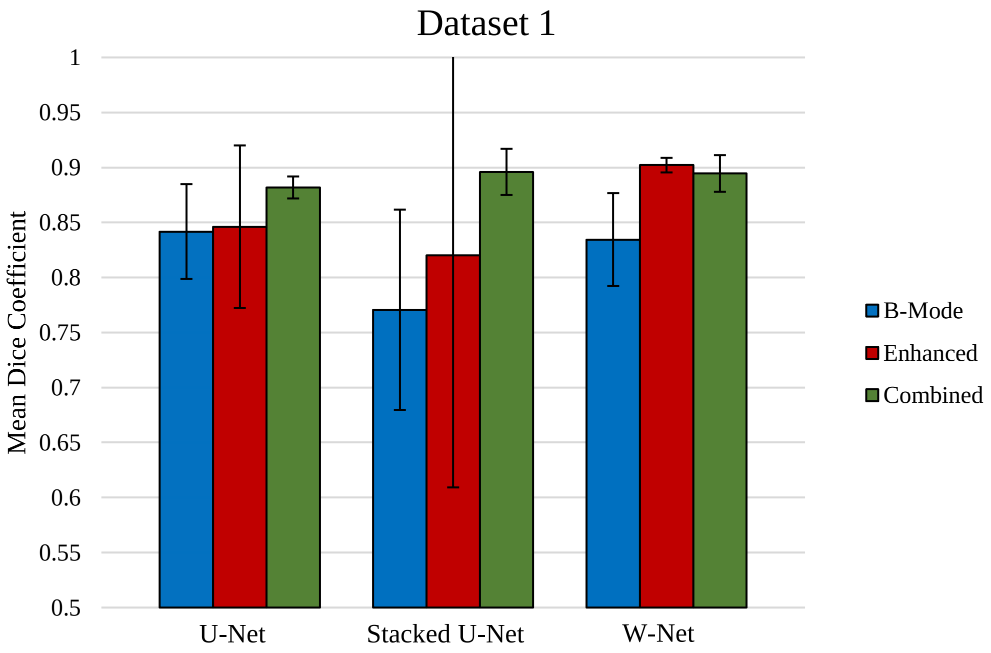


Figure 4.6: A graph showing the mean Dice Similarity Coefficient validated on Dataset #1 for the 10 networks in each combination sorted by the Network Architecture. Error bars show the standard deviation of the DSC across each set of 10 networks. One outlier in the Stacked U-Net using Enhanced images produced a large standard deviation and notably lower DSC for that set of networks. The highest average and lowest standard deviation was achieved by the W-Net.

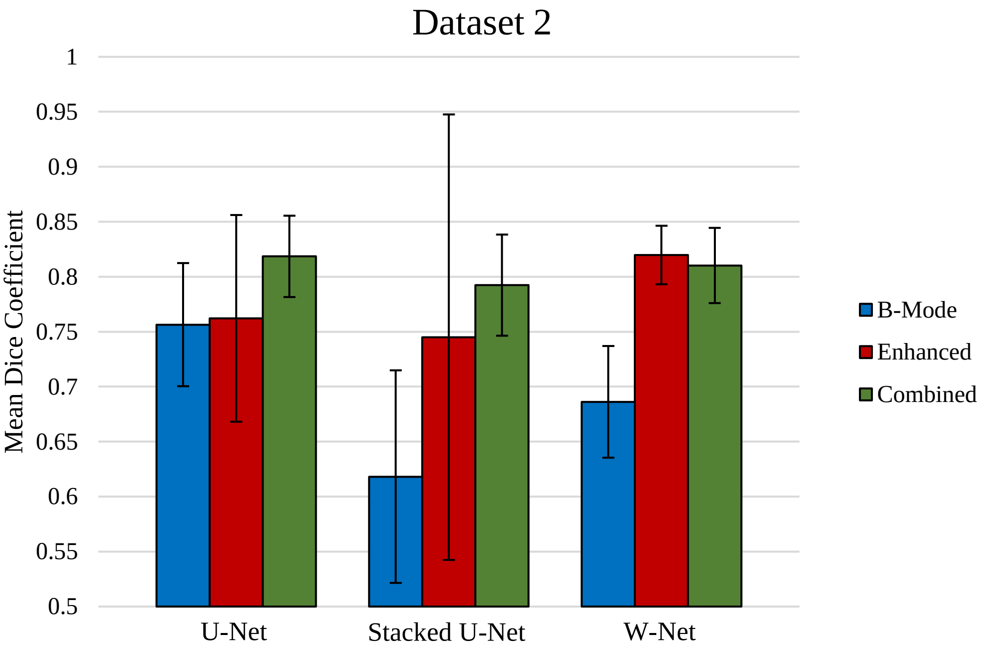


Figure 4.7: A graph showing the mean Dice Similarity Coefficient validated on Dataset #2 for the 10 networks in each combination sorted by the Network Architecture. Error bars show the standard deviation of the DSC across each set of 10 networks. One outlier in the Stacked U-Net using Enhanced images produced a large standard deviation and notably lower DSC for that set of networks. The highest average and lowest standard deviation was achieved by the W-Net.

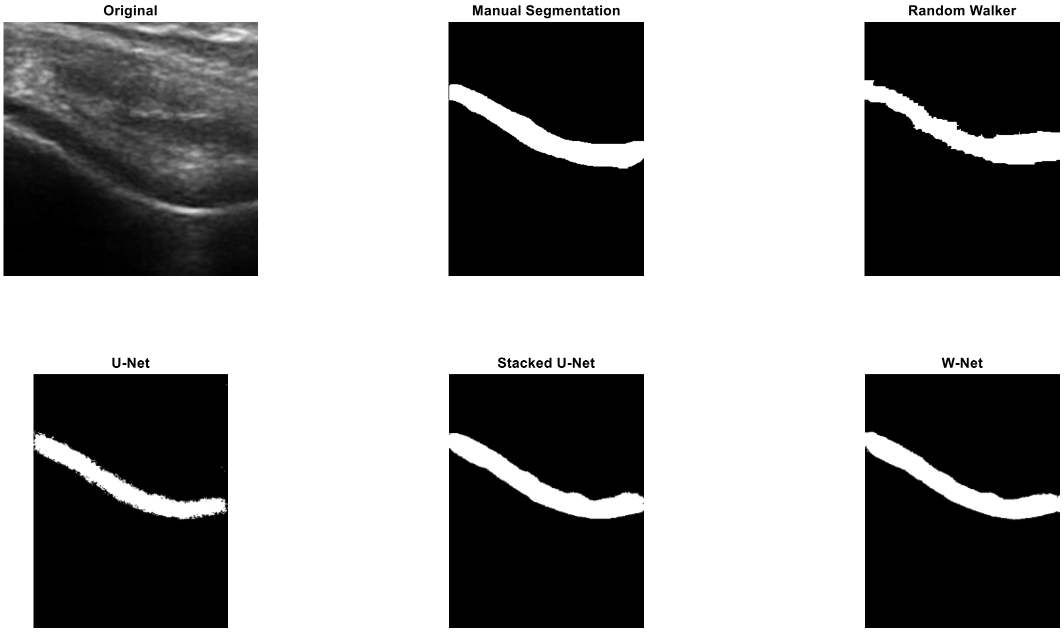


Figure 4.8: Example segmentations from dataset 1 from the best networks across the three different network architectures. The U-Net segmentation used the Enhanced image as an input. The Stacked U-Net and W-Net used the combination of both images as an input. The manual segmentation was created using the Image Segmenter in Matlab. The Random Walker shows a blocky segmentation while the Network segmentations show smoother boundaries. The DSC for the U-Net, Stacked U-Net and W-Net in this image were: 0.9209, 0.9273 and 0.9297 respectively.

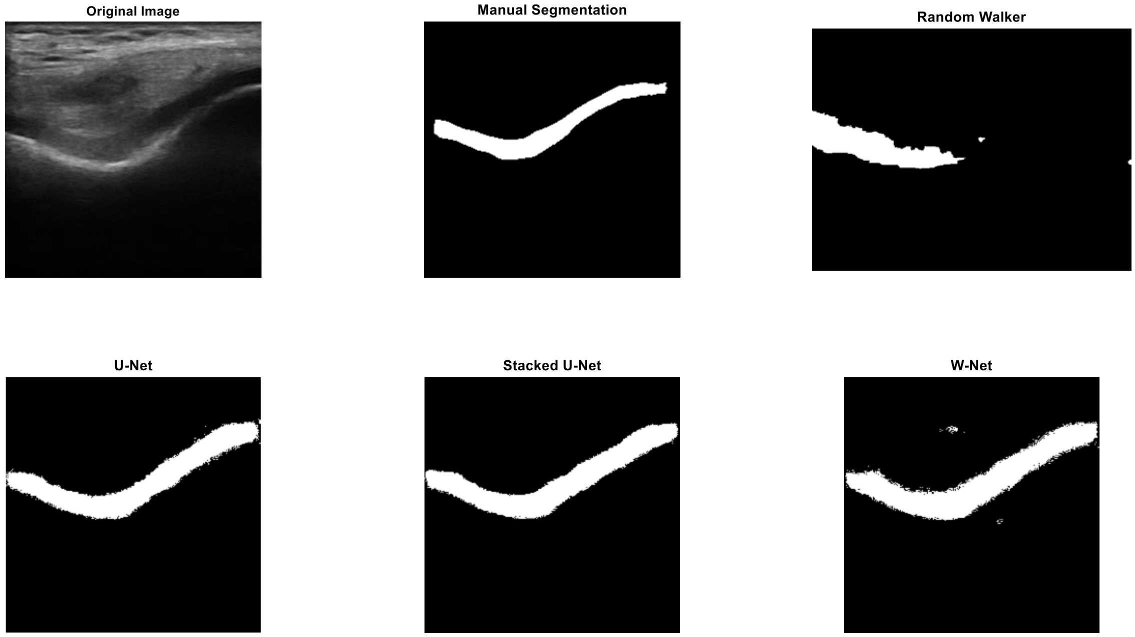


Figure 4.9: Example segmentations from dataset 2 from the best networks across the three different network architectures. The U-Net segmentation used a combination of both images as an input while the Stacked U-Net and W-Net used the Enhanced image as an input. The manual segmentation was created using the Image Segmenter in Matlab. The Random Walker segmentation in mostly unusable as the algorithm was not created for this dataset and needs different parameters to perform optimally.

4.3. Statistical Analysis

To quantify how impactful the differences in the three different Network Architectures and three different inputs into the networks, 6 standard one-way ANOVA tests were run on both datasets. The ANOVA was run on the set of DSCs that resulted from the 10-fold cross validations, with a significance level of 0.05 ($\alpha=0.05$). Three ANOVA tests were run grouping together all the DSCs from each network architecture. This left three sets of 10 DSCs, one for each input type: B-Mode, Enhanced and Combined. These three groups within each network were tested for the null hypothesis of all the mean DSCs being the same. This resulted in three p-values for each input type. This process was repeated in reverse: three ANOVA tests were run grouping together all the input types within each network architecture. This resulted in three p-values for each network architecture. All 6 ANOVA tests were done for each dataset to prevent the Yule-Simpson effect from skewing data between datasets. Table 4.4 summarizes the results of the ANOVA tests.

Table 4.4: A table summarizing the results from the one-way ANOVA tests. The ANOVA tests were done using a significance level of 0.05. For this set of data, the critical value (F_{crit}) was 3.35. To reject the null hypothesis, the displayed value of F must be larger than F_{crit}. The table is displayed to show the grouping of values on the left as identifiers as either a network type or an input type. Each network type was grouped with the DSC from the 30 networks in that category and separated it into the three input types for the ANOVA test. Each input type was grouped with the DSC from the 30 networks in that category and separated it into the three network types. The p-values that are displayed must be less than 0.05 to be considered significant. The four significant results are bolded.

Dataset 1		
Network Type	p-value	F
U-Net	0.19	1.8
Stacked U-Net	0.15	2
W-Net	**5.40E-08**	33
Input Type		
B-Mode	0.074	2.9
Enhanced	0.43	0.87
Combined	0.16	1.9
Dataset 2		
Network Type		
U-Net	0.11	2.4
Stacked U-Net	**0.026**	4.2
W-Net	**4.80E-08**	34
Input Type		
B-Mode	**0.0013**	8.6
Enhanced	0.82	0.45
Combined	0.37	1

Three results from this study were considered statistically significant. From Dataset 1, only the W-Net networks had p-values less than 0.05, while in Dataset 2 the SU-Net, W-Net and B-mode networks had p-values less than 0.05. This means that the null hypothesis of the input values having the same DSC on the W-Net can be rejected. The network architectures that use B-mode images and the SU-Net can also be considered unequal within Dataset 2 only. To further see the significance of these results, two sample t-tests were performed on the significant groups. To see the significance of the input type on the W-Net within each Dataset, 3 t-tests were completed on each combination of input types for both datasets. The t-tests performed were two sample one-tailed t-tests assuming equal variances. This produced 6 t-tests that is summarized in Table 4.5. This was repeated for the SU-Net in Dataset 2 and is shown in Table 4.6. To quantify the effects of the network architecture on the B-mode results in Dataset 2, three t-tests were done between combinations of networks in that group. The t-tests were one-tailed two sample tests assuming equal variances. The results in summarized in Table 4.7.

Table 4.5: A table showing the p-values of the 6 t-tests performed on the W-Net DSCs within the two datasets. The t-tests grouped the DSC by the type of input into the W-Net and performed one-tailed two sample t-tests assuming equal variances. The tests are represented by separating the inputs with a forward slash. The tests were performed with a significance level of 0.05 and produced four significant results which are bolded.

Test	Dataset 1	Dataset 2
B-mode/Enhanced	**2.9E-06**	**8.0E-07**
Enhanced/Combined	0.22	0.26
B-mode/Combined	**1.1E-05**	**4.9E-06**

Table 4.6: A table showing the p-values of the 6 t-tests performed on the Stacked U-Net DSCs within the two datasets. The t-tests grouped the DSC by the type of input into the Stacked U-Net and performed one-tailed two sample t-tests assuming equal variances. The tests are represented by separating the inputs with a forward slash. The tests were performed with a significance level of 0.05 and produced four significant results which are bolded. The Enhanced networks included the network with the outlier that caused a significantly lower p-value.

Test	Dataset 2
B-mode/Enhanced	0.054
Enhanced/Combined	0.25
B-mode/Combined	**6.0E-05**

Table 4.7: A table summarizing the 3 t-tests performed on the B-mode DSCs on Dataset 2. The t-tests grouped the DSC by the network architectures using B-mode images and performed one-tailed t-tests assuming equal variances. All tests were performed with a significance level of 0.05, making all three t-tests shown to be significant.

Test	p-value
U-Net / S U-Net	**0.00080**
S U-Net / W-Net	**0.039**
U-Net / W-Net	**0.0061**

The first set of t-tests performed show that using either an enhanced image or the combination of images as in input into a W-Net produced segmented images with statistically significantly higher Dice Similarity Coefficients in both datasets. This can be concluded as the average DSC for the enhanced and combined inputs is larger than that of the B-mode input W-Nets. These t-tests also show that there was no significant difference between the DSC from using an enhanced image or a combined image into a W-Net for our datasets. The second set of t-tests show that there was only a significant difference between using a B-mode input and a combined image input when using a SU-Net in Dataset 2. This shows that using a combined image was statistically better than using a B-mode image alone. The presence of the outlier in the set of enhanced images did affect the overall p-value when comparing the enhanced image to the other inputs. Specifically, if that outlier is ignored, then the p-value between the B-Mode and enhanced image inputs decreases to 4.7E-05 and the p-value between the enhanced image and the combined images lowers to 0.22. The third set of t-tests show that there is a statistically significant difference between using a U-Net SU-Net and W-Net in the second dataset. The order of the means of these DSC are: U-Net > W-Net > SU-Net. This would indicate that a U-Net produces better DSC than a W-Net which produces better DSC than a SU-Net using B-mode images with Dataset 2.

To compare using a CNN against the previously established method of using a Random Walker (RW) (Desai, 2018), a paired t-test was completed on every CNN generated. The paired t-test compared the DSC of the Neural Network against the manual segmentation and the DSC of the Random Walker against the manual segmentation of the same image. This would give a comparison of the segmentation accuracy on the same images. 90 paired t-tests were performed, and the results were arranged in Tables 4.8, 4.9 and 4.10. The p-values are shown with the null hypothesis that the CNN being tested and the RW algorithm have similar DSC. A low p-value shows that the DSC of the two algorithms are significantly different. At a confidence level of 0.05 ($\alpha = 0.05$), the p-values lower than 0.05 indicate a significant difference in the algorithms. This can, however, show that the RW algorithm performed better. To conclude which networks performed significantly better than the RW, the DSCs from Tables 4.1, 4.2, and 4.3 were compared to the DSC of the RW algorithm, 0.8481. Any CNN with a higher mean DSC than 0.8481 and a p-value of less than 0.05 can be assumed to provide a significantly better segmentation than the Random Walker. The number of these networks that are significantly better than the RW in every combination is shown in Tables 4.8, 4.9 and 4.10 as "# > RW."

Table 4.8: Table of the p-values generated from performing a paired t-test between the DSC from the U-Nets and a Random Walker algorithm on the same input images. The null hypothesis in this test is that the DSC from both algorithms are the same; a low p-value indicates a significant difference between the two. The p-values that also corresponded to a higher mean DSC are in bold to indicate that the DSC of this network are significantly larger than that of the Random Walker. The total number of these networks for each input type is shown in the last row of the table.

U-Net			
Net Index	B-Mode	Enhanced	Combined
1	**2.4 E-07**	0.87	**0.00032**
2	4.5 E-29	1.4 E-41	**0.0027**
3	**0.00026**	**0.0042**	**1.4 E-05**
4	**0.049**	**0.00031**	**0.00023**
5	4.7 E-06	0.148	**1.8 E-06**
6	**0.00053**	**0.00012**	0.31
7	0.12	0.31	0.47
8	2.2 E-07	**2.5 E-06**	**0.00010**
9	0.011	**2.9 E-12**	0.85
10	**0.028**	1.0 E-12	0.22
# >RW	5	5	6

Table 4.9: Table of the p-values generated from performing a paired t-test between the DSC from the Stacked U-Nets and a Random Walker algorithm on the same input images. The null hypothesis in this test is that the DSC from both algorithms are the same; a low p-value indicates a significant difference between the two. The p-values that also corresponded to a higher mean DSC are in bold to indicate that the DSC of this network are significantly larger than that of the Random Walker. The total number of these networks for each input type is shown in the last row of the table.

Stacked U-Net			
Net Index	B-Mode	Enhanced	Combined
1	1.2 E-37	2.4 E-65	**2.8 E-10**
2	**0.0014**	**1.3 E-14**	**7.9 E-17**
3	0.13	**4.8 E-05**	**1.1 E-15**
4	**2.6 E-15**	**3.4 E-11**	**1.5 E-07**
5	2.4 E-17	**1.3 E-12**	**1.1 E-15**
6	1.9 E-14	**7.5 E-18**	0.0015
7	1.1 E-29	**1.9 E-05**	**3.4 E-09**
8	1.2 E-26	**4.5 E-17**	**2.6 E-13**
9	**4.7 E-12**	**4.0 E-16**	**3.8 E-09**
10	0.0069	1.0 E-08	**4.5 E-11**
# >RW	3	8	9

Table 4.10: Table of the p-values generated from performing a paired t-test between the DSC from the W-Nets and a Random Walker algorithm on the same input images. The null hypothesis in this test is that the DSC from both algorithms are the same; a low p-value indicates a significant difference between the two. The p-values that also corresponded to a higher mean DSC are in bold to indicate that the DSC of this network are significantly larger than that of the Random Walker. The total number of these networks for each input type is shown in the last row of the table.

W-Net			
Net Index	B-Mode	Enhanced	Combined
1	7.7 E-11	**3.6 E-14**	**1.1 E-13**
2	0.00028	**9.6 E-13**	**2.5 E-10**
3	3.5 E-11	**1.1 E-16**	**4.7 E-06**
4	1.1 E-09	**1.7 E-10**	**4.2 E-12**
5	1.2 E-15	**1.6 E-13**	**1.2 E-09**
6	6.1 E-20	**3.1 E-14**	**3.0 E-11**
7	0.83	**1.1 E-11**	**2.8 E-09**
8	**5.0 E-07**	**5.1 E-15**	**2.6 E-11**
9	6.4 E-31	**7.2 E-11**	**1.1 E-13**
10	2.2 E-11	**1.0 E-15**	**4.3 E-13**
# >RW	1	10	10

In total, 16 of the 30 U-Nets, 20 of the 30 SU-Nets, and 21 of the 30 W-Nets performed better than the RW. 9 of the 30 B-Mode inputs, 23 of the 30 Enhanced input and 25 of the 30 Combined input networks performed better than the Random Walker.

Chapter 5

Discussion & Conclusions

The goal of this study was to see the effects of using the three different network architectures of the: U-Net, Stacked U-Net and W-Net while testing the effect of using local phase enhancement on the B-Mode knee images on segmentation accuracy. The following sections will evaluate how the generated CNNs performed in comparison to each other and a Random Walker that was established to be the best segmentation method for the dataset used.

5.1 Quality of Segmentation

Dice Similarity Coefficient is a widely established method of evaluating semantic segmentation algorithms. It allows a numerical value to be assigned to the accuracy of a segmentation. It is, however, a naïve measurement of the accuracy of medical image segmentation. A DSC does not take any biological landmarks or location of errors into calculation and cannot give a quality measurement to segmentation algorithms. DSC has limitations as a quality measurement but has a role as quantifying results. It is, however, difficult to give a visual score to segmentations. For this reason, artifacts in segmentation results will be mentioned in entirety. A common artifact that was segmented by the Neural Networks was the presence of a hypoechoic ridge above the knee cartilage. This artifact can most easily be seen in Figure 4.3 in the B-Mode segmentation. This artifact is produced by the patellar tendon and appears differently depending on the patient. This artifact is difficult to avoid in some automated systems of segmentation such as Watershed and Random Walker that use automated seed points to grow outwards (Desai, 2018). Another artifact in segmentations that appeared to be exclusive to the U-Nets in this study is noisy edges. Figures 4.8 and 4.9 show examples of this artifact, with the edges of the segmentations appearing powdery and rough. This artifact persisted through lower and higher thresholding values of the probability, showing that the U-Nets are not confident in finding the edges in the segmentation. An explanation for this behavior is that the U-Nets were overfit to the training data, but this artifact appeared when predicting training data as well. This edge blurriness can instead be explained by the network training algorithm not punishing these blurry edges significantly enough given the U-Net architecture. The addition of more weights and biases in the SU-Net and W-Net architecture would work to smooth out these edges by having more total learnables. Determination of a ground truth model can be problematic for training neural networks. In this study, a manual segmentation of an ultrasound was used as the ground truth for training and validating the accuracy of the networks. This model is prone to human error and inconsistency around segmentation. A goal of this study was to segment the cartilage of the knee in ultrasound images to avoid the noise that damages the performance in other automated systems. This same noise also leads to difficulty in segmentation and can interfere with network training and can skew the absolute DSC, but not the relative DSC between groups within the study. Every figure shown with a manual segmentation is not perfect: rough lines and spikes in straight lines from using a mouse to segment can lead to imperfect segmentations. These imperfections are amplified through training with the low number of dataset images in this study. It is possible that a neural network can produce better segmentations than a manual segmentation, but the DSC would be lower in validation. Averaging together multiple expert segmentations can improve the accuracy of the ground truth but can also cause smoothing of rougher edges in the true segmentation. Determining an absolute ground truth would be impossible without invasive surgery to physically measure the cartilage thickness of each patient, which is infeasible for this study. Changing the validation metric to a more physical value, such as median Joint Space Width, can improve the ground truth for validation, but it would not be able to be used for error backpropagation in training the networks. This dichotomy in training metric and validation metric could produce networks with high DSC and low JSW similarity. Combining the ultrasound images with an image from another modality can also improve the ground truth accuracy. If two modalities are being used in this way, however, it would be better to combine the inputs of both imaging modalities into the network, which can lead to problems with having enough unique images for training without overfit. This study used the current “golden standard” in Ultrasound imaging, manual segmentation, but must acknowledge that it is imperfect.

5.2 Impact of Enhancement

One novel part of this study is using an enhanced image as an input into the Neural Networks. A fair criticism of this decision is that any consistent transformation of an input into a network can be reconstructed with any Convolutional Neural Network given enough learnables (Russell, 1995). Given a large enough network, the enhancement algorithm used in this study can be learned by a neural network if it is trained to create it. With the limited amount of training images, however, this algorithm cannot be expected to form from networks with the amount of learnables used in this study. Within the results from this study, using an enhanced image or a combination of the B-mode image and the enhanced image was only significantly better when using a W-Net compared to the other networks as shown in Table 4.5. This table also shows that there was not a significant difference between using the enhanced image and using the combination image as an input into the W-Net. These results would imply that the enhanced image was more impactful than the presence of the B-Mode image in the combination input for W-Nets. This is important because networks that used a combination of inputs had more learnables to account for the input. These networks had 576 more learnables, which should give an advantage in the amount of details it would be able to identify with perfect training. This means that the effect of these extra learnables was overshadowed by the effect of using an enhanced image as an input. Using an enhanced image or combination as an input into the networks compared to the RW algorithm in Tables 4.8, 4.9 and 4.10 always show the same ore more networks that produced significantly higher DSCs. This difference, again, is most present in the W-Net where the number of outperforming networks increased from 10% of the networks to 100%. Overall, it appears that a more complicated network architecture such as the W-Net benefits more from the image enhancement than a simpler network such as a U-Net. The W-Net has many more learnables than the U-Net that need to be updated to reach an ideal segmentation. Given the same number of training iterations between the two, it would make sense that the W-Net would reach the stage of training that is focusing on finer details earlier in training if given the enhanced image as an input. Moreover, this trend is not seen in the SU-Net to the same degree because of the lack of the crosstalk between the main arcs in the architecture. The crosstalk in the W-Net may allow a more direct connection to the segmentation layer and prevent the important details from being lost in the forward propagation. The effect of using an enhanced image in a SU-Net is less clear. The presence of an outlier in the networks caused skewing of the p-values, as discussed in Section 4.3. If the outlier is ignored, then it can be concluded that the SU-Net performed better in Dataset 2 when using an enhanced image or a combination of images instead of the B-mode image alone. With the outlier, however, this can only be extended to the combination of images. Since this only applies to Dataset 2, the images less similar to the images the network was trained on, it would again imply that the enhanced images improved the robustness of the final network after training and allowed for easier segmentation when the inputs are enhanced.

5.3 Impact of Network Architecture

When looking at the effect the Network Architecture had on the segmentations, it is difficult to draw many conclusions. Overall, the comparisons between networks along the same inputs did not give rise to many significant results between each other. The only significant results were using a B-Mode input on Dataset 2 as shown in Table 4.8. This showed that when using a B-mode image, the U-Net performed better than the W-Net, which performed better than the SU-Net. This would imply that the U-Net was the best network for segmentation of less similar images than what it was trained on but underperformed on images like that which it was trained on. This tradeoff of accuracy in a specific image type for accuracy in robustness of the network can be justified when training networks as templates for other uses. That is, using the trained U-Net in deep learning methods where the network would be retrained briefly on a newer dataset. This is niche but can have uses in general segmentation systems that would see many different types of images.

5.4 Comparison to Random Walker

The Random Walker algorithm was chosen as the method to compare against the Neural Networks in this study, as it was previously established as the best method of segmentation compared to both Watershed and Graph-Cut algorithms on this specific dataset (Desai, 2018). The Neural Networks in this study were shown to significantly outperform the Random Walker segmentations when using either an enhanced image or combined image as an input, as shown in Tables 4.8 4.9, and 4.10. Both the SU-Net and the W-Net both underperformed when using only the B-Mode image but improved greatly when using the enhanced and combined images. This is contrary to the U-Net, which saw similar performance across all inputs when compared to the RW. The only combination of networks that did not have a majority be statistically better DSC than the RW were W-Nets and SU-Nets with the B-mode as an input. Otherwise, every other network was shown to be better. The U-Net had at least 50% of the networks to be better across all inputs, the SU-Net had 80% and 90% better for using the Enhanced and Combined input respectively, and the W-Net with the enhanced and combined input had 90% and 100% of the networks be better.

5.5 Conclusions

The segmentation of Knee cartilage using three different Neural Network architectures: U-Net, SU-Net and W-Net, using B-mode, local phase filtering and bone shadow enhanced images and a combination of the two images as inputs for each were compared in this study. The performance of these networks was compared against the best automated segmentation method established on this dataset, the Random Walker algorithm.

The enhancement of training and testing images in the Neural Networks in this study were compared to similar networks that used pre-enhanced B-Mode images. It was shown that the enhanced images had a significant impact on the performance of the W-Net when comparing the Dice Similarity Coefficients (DSC) against other networks and inputs in this study. This shows that the W-Net benefitted from having the enhanced images. The different network architectures were shown to be similar to each other in the datasets in this study. The only significant difference in DSC was found on networks that used the B-mode images as an input and only in the dataset that was less similar than the images the networks were trained on. This showed that the U-Net outperformed the W-Net, and the W-Net outperformed the SU-Net. When comparing the Neural Networks in this study to the previously established best automatic segmentation algorithm: Random Walker, the networks outperformed when using the enhanced images. In this metric, the SU-Net and W-Net also outperformed the U-Net when using the enhanced images as an input. The U-Net, however, performed better when using the B-Mode images alone. Quantitatively, the Neural Networks outperformed the Random Walker in most cases presented in this study, and qualitatively produced much smoother segmentations than that of the Random Walker. It is difficult to conclude which Neural Network was the best performing, as different metrics contradict each other, but the U-Net was the most consistently performing network with a lower storage size and faster training and computation time. The other two networks had examples where they outperformed the U-Net, but this tradeoff with computation time and storage may be more useful in some clinical systems that attempt to perform real-time segmentation. The promise of mass throughput by using Neural Networks can allow many automated segmentations of knee cartilage in rapid succession and give rise to newer technologies that can assist both clinical and research alike.

5.6 Challenges

Many challenges in this study were faced with the nature of Neural Networks and the ever-expanding technologies for creating and training them. Thankfully, the use of a GPU instead of a CPU for the training of the networks shorten the total training time per network from approximately 6 hours to 30 minutes. Training 90 networks with these times became much easier with advancements in CUDA® cores and parallel computing. Moreover, the choices of parameters for training and attempting to prevent problems such as gradient explosion during training was tricky to achieve across the different network architectures and inputs given the limited RAM available for the GPU. Finally, the computation time for the Neural networks is relatively low, averaging 0.03 seconds for U-Nets and 0.05 seconds for SU-Net and W-Nets. This is, however, overshadowed by the time it takes to enhance the images to use in these networks, which takes approximately 0.2 seconds. Overall, this should not affect most cases where this algorithm is used, but in real-time systems may be noticeable when combined with graphical rendering lag.

5.7 Future Work

This study looked specifically at the ability of CNNs to segment knee cartilage from US. This implementation had limitations that can be investigated and investigated with further studies to continue the optimization of knee cartilage measurements and bring these technologies to the clinical space.

5.7.1 Dice Similarity Coefficient Scaled Neural Network Training

One major limitation of this study was the CNN training using Stochastic Gradient Descent. This method of error backpropagation is widely accepted in many different uses for CNNs but is limited to a measurement of accuracy to determine a dynamic learning rate across the training process. In the case of this study, training accuracy was represented as a percentage of correct guesses from the CNN in sorting pixels into the foreground and background. Scaling this accuracy by the Dice Similarity Coefficient during training can prevent cases of marginal increases in pixel accuracy leading to no change in the Dice Coefficient.

5.7.2 Real-Time Knee Cartilage Segmentation

The CNNs generated in this study allowed for relatively low segmentation times from the Ultrasound images in the datasets. This can be extended to allow real-time segmentation of the cartilage that can aid with transducer placement during measurement and allow a more accurate representation of the B-mode image a clinician is attempting to capture.

5.7.3 Comparison of Ultrasound and MRI Knee Cartilage Segmentations

MR and ultrasound both allow for soft tissue imaging of the knee for measuring cartilage. There have not been any major studies looking at the differences between the measurements from ultrasound images and MRI from the same patients to see the accuracy differences between the two modalities. This comparison can show the clinical accuracy of ultrasound as a viable method of measuring knee cartilage.

5.8 Thesis Contributions

The main contributions of this thesis are summarized as:

1) The comparison between the U-Net, Stacked U-Net and W-Net architectures be in an ultrasound imaging application. This study showed that the choice of neural network architecture had limited effects on the final segmentations generated. The U-Net performed more effectively in the dataset that was more different than the training set, showing that it avoided overfit and may have been more robust.
2) The use of local phase enhancement of B-Mode ultrasound images for improving the accuracy and consistency of Convolutional Neural Networks. The use of enhanced images greatly increased the segmentation power of the CNNs and increased both the quantitative performance and quality of segmentation. Fusing the enhanced images and the B-mode images at the beginning of the CNN also proved to increase performance.
3) The comparison of using a Convolutional Neural Network against a Random Walker for segmenting knee cartilage from Ultrasound images. The CNNs generated were shown to outperform the Random Walker when using the enhanced images or when using a more complex network architecture.

References

Alom, M., Yakopcic, C., Hasan, M., Taha, T., & Asari, V. (2019). Recurrent residual U-Net for medical image segmentation. *Journal Of Medical Imaging*, *6*(01), 1. https://doi.org/10.1117/1.jmi.6.1.014006

Ambellan, F., Tack, A., Ehlke, M., & Zachow, S. (2019). Automated segmentation of knee bone and cartilage combining statistical shape knowledge and convolutional neural networks: Data from the Osteoarthritis Initiative. *Medical image analysis*, *52*, 109-118.

Amorim, P., Moraes, T., Silva, J., & Pedrini, H. (2018). 3D Adaptive Histogram Equalization Method for Medical Volumes. *Proceedings Of The 13Th International Joint Conference On Computer Vision, Imaging And Computer Graphics Theory And Applications*. https://doi.org/10.5220/0006615303630370

Antico, M., Sasazawa, F., Dunnhofer, M., Camps, S. M., Jaiprakash, A. T., Pandey, A. K., ... & Fontanarosa, D. (2020). Deep learning-based femoral cartilage automatic segmentation in ultrasound imaging for guidance in robotic knee arthroscopy. *Ultrasound in Medicine & Biology*, *46*(2), 422-435.

Antony, J., McGuinness, K., Moran, K., & O'Connor, N. E. (2017, July). Automatic detection of knee joints and quantification of knee osteoarthritis severity using convolutional neural networks. In *International conference on machine learning and data mining in pattern recognition* (pp. 376-390). Springer, Cham.

Baka, N., Leenstra, S., & van Walsum, T. (2017). Random Forest-Based Bone Segmentation in Ultrasound. *Ultrasound In Medicine & Biology*, *43*(10), 2426-2437. https://doi.org/10.1016/j.ultrasmedbio.2017.04.022

Barr, R., Hindi, & Peterson. (2013). Artifacts in diagnostic ultrasound. *Reports In Medical Imaging*, 29. https://doi.org/10.2147/rmi.s33464

Beattie, K. A., Duryea, J., Pui, M., O'Neill, J., Boulos, P., Webber, C. E., Eckstein, F., & Adachi, J. D. (2008). Minimum joint space width and tibial cartilage morphology in the knees of healthy individuals: a cross-sectional study. *BMC musculoskeletal disorders*, *9*, 119. https://doi.org/10.1186/1471-2474-9-119

Benzarti, F., & Amiri, H. (2013). Speckle noise reduction in medical ultrasound images. *arXiv preprint arXiv:1305.1344*.

Bitton, R. (2009). The economic burden of osteoarthritis. *The American journal of managed care*, *15*(8 Suppl), S230-5.

Brandt, K. D., Fife, R. S., Braunstein, E. M., & Katz, B. (1991). Radiographic grading of the severity of knee osteoarthritis: relation of the Kellgren and Lawrence grade to a grade based on joint space narrowing, and correlation with arthroscopic evidence of articular cartilage degeneration. *Arthritis and rheumatism*, *34*(11), 1381–1386. https://doi.org/10.1002/art.1780341106

Boukerroui, D., Noble, J. A., & Brady, M. (2004). On the choice of band-pass quadrature filters. *Journal of Mathematical Imaging and Vision*, *21*(1-2), 53-80.

Carballido-Gamio, J., Bauer, J. S., Stahl, R., Lee, K. Y., Krause, S., Link, T. M., & Majumdar, S. (2008). Inter-subject comparison of MRI knee cartilage thickness. *Medical image analysis*, *12*(2), 120–135. https://doi.org/10.1016/j.media.2007.08.002

Chen, S. D., & Ramli, A. R. (2003). Minimum mean brightness error bi-histogram equalization in contrast enhancement. *IEEE transactions on Consumer Electronics, 49*(4), 1310-1319.

Chen, W., Zhang, Y., He, J., Qiao, Y., Chen, Y., Shi, H., & Tang, X. (2018). W-net: Bridged U-net for 2D Medical Image Segmentation. *ArXiv, abs/1807.04459.*

Çiçek, Ö., Abdulkadir, A., Lienkamp, S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. *Medical Image Computing And Computer-Assisted Intervention – MICCAI 2016*, 424-432. https://doi.org/10.1007/978-3-319-46723-8_49

Desai, P. R., & Hacihaliloglu, I. (2018, April). Enhancement and automated segmentation of ultrasound knee cartilage for early diagnosis of knee osteoarthritis. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)* (pp. 1471-1474). IEEE.

Desai, P., & Hacihaliloglu, I. (2019). Knee-cartilage segmentation and thickness measurement from 2D ultrasound. *Journal of Imaging*, *5*(4), 43.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, *26*(3), 297-302.

Dong, H., Yang, G., Liu, F., Mo, Y., & Guo, Y. (2017). Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks. *Communications In Computer And Information Science*, 506-517. https://doi.org/10.1007/978-3-319-60964-5_44

Dunnhofer, M., Antico, M., Sasazawa, F., Takeda, Y., Camps, S., Martinel, N., ... & Fontanarosa, D. (2020). Siam-U-Net: encoder-decoder siamese network for knee cartilage tracking in ultrasound images. *Medical Image Analysis*, *60,* 101631.

Duryea, J., Li, J., Peterfy, C. G., Gordon, C., & Genant, H. K. (2000). Trainable rule-based algorithm for the measurement of joint space width in digital radiographic images of the knee. *Medical physics*, *27*(3), 580–591. https://doi.org/10.1118/1.598897

Fairbank T. J. (1948). Knee joint changes after meniscectomy. *The Journal of bone and joint surgery. British volume*, *30B*(4), 664–670.
Faisal, A., Ng, S., Goh, S., & Lai, K. (2017). Knee cartilage segmentation and thickness computation from ultrasound images. *Medical & Biological Engineering & Computing*, *56*(4), 657-669. https://doi.org/10.1007/s11517-017-1710-2

Galli, M., De Santis, V., & Tafuro, L. (2003). Reliability of the Ahlbäck classification of knee osteoarthritis. *Osteoarthritis and cartilage, 11*(8), 580–584. https://doi.org/10.1016/s1063-4584(03)00095-5

Gilboa, G., Sochen, N., & Zeevi, Y. Y. (2004). Image enhancement and denoising by complex diffusion processes. *IEEE transactions on pattern analysis and machine intelligence*, *26*(8), 1020-1036.

Gornale, S. S., Patravali, P. U., & Manza, R. R. (2016). Detection of osteoarthritis using knee X-ray image analyses: a machine vision based approach. *International Journal of Computer Applications*, *145*(1).

Hacihaliloglu, I., Abugharbieh, R., Hodgson, A. J., & Rohling, R. N. (2009). Bone surface localization in ultrasound using image phase-based features. *Ultrasound in medicine & biology*, *35*(9), 1475-1487.

Hayashi, D., Roemer, F., & Guermazi, A. (2016). Imaging for osteoarthritis. *Annals Of Physical And Rehabilitation Medicine*, *59*(3), 161-169. https://doi.org/10.1016/j.rehab.2015.12.003

Hefti, F., Müller, W., Jakob, R. P., & Stäubli, H. U. (1993). Evaluation of knee ligament injuries with the IKDC form. *Knee surgery, sports traumatology, arthroscopy : official journal of the ESSKA*, *1*(3-4), 226–234. https://doi.org/10.1007/bf01560215

Henderson, R. E., Walker, B. F., & Young, K. J. (2015). The accuracy of diagnostic ultrasound imaging for musculoskeletal soft tissue pathology of the extremities: a comprehensive review of the literature. *Chiropractic & manual therapies*, *23*(1), 31.

Hong-Seng, G., Sayuti, K. A., & Karim, A. H. A. (2017). Investigation of random walks knee cartilage segmentation model using inter-observer reproducibility: Data from the osteoarthritis initiative. *Bio-medical materials and engineering*, *28*(2), 75-85.

Hossain, M., Lai, K., Pingguan-Murphy, B., Hum, Y., Mohd Salim, M., & Liew, Y. (2014). Contrast enhancement of ultrasound imaging of the knee joint cartilage for early detection of knee osteoarthritis. *Biomedical Signal Processing And Control*, *13*, 157-167. https://doi.org/10.1016/j.bspc.2014.04.008

Hossain, B., Pingguan-Murphy, B., Hum, Y.C., & Lai, K.W. (2015). Contrast Enhancement of Ultrasound Image of Knee Joint Cartilage by Using Multipurpose Beta Optimized Recursive Bi-Histogram Equalization Method. *ICIS 2015*.

Irrgang, J. J., Anderson, A. F., Boland, A. L., Harner, C. D., Neyret, P., Richmond, J. C., Shelbourne, K. D., & International Knee Documentation Committee (2006). Responsiveness of the International Knee Documentation Committee Subjective Knee Form. *The American journal of sports medicine*, *34*(10), 1567–1573. https://doi.org/10.1177/0363546506288855

Kashyap, S., Oguz, I., Zhang, H., & Sonka, M. (2016, October). Automated segmentation of knee MRI using hierarchical classifiers and just enough interaction based learning: Data from osteoarthritis initiative. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 344-351). Springer, Cham.

Keen, H. I., & Conaghan, P. G. (2009). Ultrasonography in osteoarthritis. *Radiologic Clinics*, *47*(4), 581-594.

Kellgren, J. H., & Lawrence, J. S. (1957). Radiological assessment of osteo-arthrosis. *Annals of the rheumatic diseases*, *16*(4), 494–502. https://doi.org/10.1136/ard.16.4.494

Kim, M., & Chung, M. G. (2008). Recursively separated and weighted histogram equalization for brightness preservation and contrast enhancement. *IEEE Transactions on Consumer Electronics*, *54*(3), 1389-1397.

Kim, Y. T. (1997). Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE transactions on Consumer Electronics*, *43*(1), 1-8.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kohn, M. D., Sassoon, A. A., & Fernando, N. D. (2016). Classifications in brief: Kellgren-Lawrence classification of osteoarthritis.

Kompella, G., Antico, M., Sasazawa, F., Jeevakala, S., Ram, K., Fontanarosa, D., ... & Sivaprakasam, M. (2019, July). Segmentation of femoral cartilage from knee ultrasound images using mask R-CNN. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 966-969). IEEE.

Lee, J. G., Gumus, S., Moon, C. H., Kwoh, C. K., & Bae, K. T. (2014). Fully automated segmentation of cartilage from the MR images of knee using a multi-atlas and local structural analysis method. *Medical physics*, *41*(9), 092303.

Lepuesne, M., Brandt, K., Bellamy, N., Moskowitz, R., Menkes, C. J., & Pelletier, J. P. (1994). Guidelines for testing slow acting drugs in osteoarthritis. *J Rheumatol*, *21*, 65-73.

Michailovich, O. V., & Tannenbaum, A. (2006). Despeckling of medical ultrasound images. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, *53*(1), 64–78. https://doi.org/10.1109/tuffc.2006.1588392

Milletari, F., Navab, N., & Ahmadi, S. (2016). V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. *2016 Fourth International Conference On 3D Vision (3DV)*. https://doi.org/10.1109/3dv.2016.79

Nagao, N., Tachibana, T. & Mizuno, K. The rotational angle in osteoarthritic knees. *International Orthopaedics SICOT* **22,** 282–287 (1998). https://doi.org/10.1007/s002640050261

Perona, P., & Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on pattern analysis and machine intelligence*, *12*(7), 629-639.

Prasoon, A., Petersen, K., Igel, C., Lauze, F., Dam, E., & Nielsen, M. (2013, September). Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. In *International conference on medical image computing and computer-assisted intervention* (pp. 246-253). Springer, Berlin, Heidelberg.

Razek, A. A. K. A., Fouda, N. S., Elmetwaley, N., & Elbogdady, E. (2009). Sonography of the knee joint. *Journal of ultrasound*, *12*(2), 53-60.

Riecke, B., Christensen, R., Torp-Pedersen, S., Boesen, M., Gudbergsen, H., & Bliddal, H. (2014). An ultrasound score for knee osteoarthritis: a cross-sectional validation study. *Osteoarthritis And Cartilage*, *22*(10), 1675-1691. https://doi.org/10.1016/j.joca.2014.06.020

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *Lecture Notes In Computer Science*, 234-241. https://doi.org/10.1007/978-3-319-24574-4_28

Russell, S. and Norvig, P., (1995). *Artificial Intelligence: A Modern Approach*. 1st ed. Englewood Cliffs: Prentice-Hill, pp.578-584.

Saba, T., Rehman, A., Mehmood, Z., Kolivand, H., & Sharif, M. (2018). Image Enhancement and Segmentation Techniques for Detection of Knee Joint Diseases: A Survey. *Current Medical Imaging Reviews*, *14*(5), 704-715. https://doi.org/10.2174/1573405613666170912164546

Scheller, G., Sobau, C., & Bülow, J. U. (2001). Arthroscopic partial lateral meniscectomy in an otherwise normal knee: Clinical, functional, and radiographic results of a long-term follow-up study. *Arthroscopy : the journal of arthroscopic & related surgery : official publication of the Arthroscopy Association of North America and the International Arthroscopy Association*, *17*(9), 946–952. https://doi.org/10.1053/jars.2001.28952

Schiphof, D., Boers, M., & Bierma-Zeinstra, S. M. (2008). Differences in descriptions of Kellgren and Lawrence grades of knee osteoarthritis. *Annals of the rheumatic diseases*, *67*(7), 1034–1036. https://doi.org/10.1136/ard.2007.079020
Schroeder-Boersch, H., Töws, P., & Jani, L. (1998). Reproduzierbarkeit von radiologischen Arthrosemerkmalen. Evaluierung eines Scores zur radiologischen Klassifikation von degenerativen Veränderungen bei Gonarthrose [Reproducibility of radiologic markers of osteoarthritis. Evaluating a score for radiologic classification of degenerative changes in osteoarthritis of the knee joint]. *Zeitschrift fur Orthopadie und ihre Grenzgebiete*, *136*(4), 293–297. https://doi.org/10.1055/s-2008-1053740

Segal, N. A., Nevitt, M. C., Lynch, J. A., Niu, J., Torner, J. C., & Guermazi, A. (2015). Diagnostic performance of 3D standing CT imaging for detection of knee osteoarthritis features. *The Physician and sportsmedicine*, *43*(3), 213-220

Shah, S., Ghosh, P., Davis, L. S., & Goldstein, T. (2018). Stacked U-Nets: a no-frills approach to natural image segmentation. *arXiv preprint arXiv:1804.10343*.

Sim, K. S., Tso, C. P., & Tan, Y. Y. (2007). Recursive sub-image histogram equalization applied to gray scale images. *Pattern Recognition Letters*, *28*(10), 1209-1221.

Sun, T., Chen, Z., Yang, W., & Wang, Y. (2018, June). Stacked U-Nets With Multi-Output for Road Extraction. In *CVPR Workshops* (pp. 202-206).

Swanson, M. S., Prescott, J. W., Best, T. M., Powell, K., Jackson, R. D., Haq, F., & Gurcan, M. N. (2010). Semi-automated segmentation to assess the lateral meniscus in normal and osteoarthritic knees. *Osteoarthritis and cartilage*, *18*(3), 344-353.

Tang, Z., Peng, X., Geng, S., Zhu, Y., & Metaxas, D. N. (2018). CU-net: coupled U-nets. *arXiv preprint arXiv:1808.06521*.

Urish, K. L., Keffalas, M. G., Durkin, J. R., Miller, D. J., Chu, C. R., & Mosher, T. J. (2013). T2 texture index of cartilage can predict early symptomatic OA progression: data from the osteoarthritis initiative. *Osteoarthritis and cartilage*, *21*(10), 1550-1557.

Wallace, I. J., Worthington, S., Felson, D. T., Jurmain, R. D., Wren, K. T., Maijanen, H., ... & Lieberman, D. E. (2017). Knee osteoarthritis has doubled in prevalence since the mid-20th century. *Proceedings of the National Academy of Sciences*, *114*(35), 9332-9336.

Wang, C., MacGillivray, T., Macnaught, G., Yang, G., & Newby, D. (2018). A two-stage 3D Unet framework for multi-class segmentation on full resolution image. *arXiv preprint arXiv:1804.04341*.

Wang, P., Patel, V. M., & Hacihaliloglu, I. (2018, September). Simultaneous segmentation and classification of bone surfaces from ultrasound using a multi-feature guided CNN. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 134-142). Springer, Cham.

Wang, Y., Chen, Q., & Zhang, B. (1999). Image enhancement based on equal area dualistic sub-image histogram equalization method. *IEEE Transactions on Consumer Electronics*, *45*(1), 68-75.

Xia, X., & Kulis, B. (2017). W-net: A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv:1711.08506*.

Yang, G., & Schoenholz, S. S. (2018). Deep mean field theory: Layerwise variance and width variation as methods to control gradient explosion.

Yongjian Yu, & Acton, S. (2002). Speckle reducing anisotropic diffusion. *IEEE Transactions On Image Processing*, *11*(11), 1260-1270. https://doi.org/10.1109/tip.2002.804276

Yu, J., Tan, J., & Wang, Y. (2010). Ultrasound speckle reduction by a SUSAN-controlled anisotropic diffusion method. *Pattern recognition*, *43*(9), 3083-3092.

Zanella, M., Kohlmann, O., & Ribeiro, A. (2001). Treatment of Obesity Hypertension and Diabetes Syndrome. *Hypertension*, *38*(3), 705-708. https://doi.org/10.1161/01.hyp.38.3.705

Zou, K. H., Warfield, S. K., Bharatha, A., Tempany, C. M., Kaus, M. R., Haker, S. J., ... & Kikinis, R. (2004). Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. *Academic radiology*, *11*(2), 178-189.

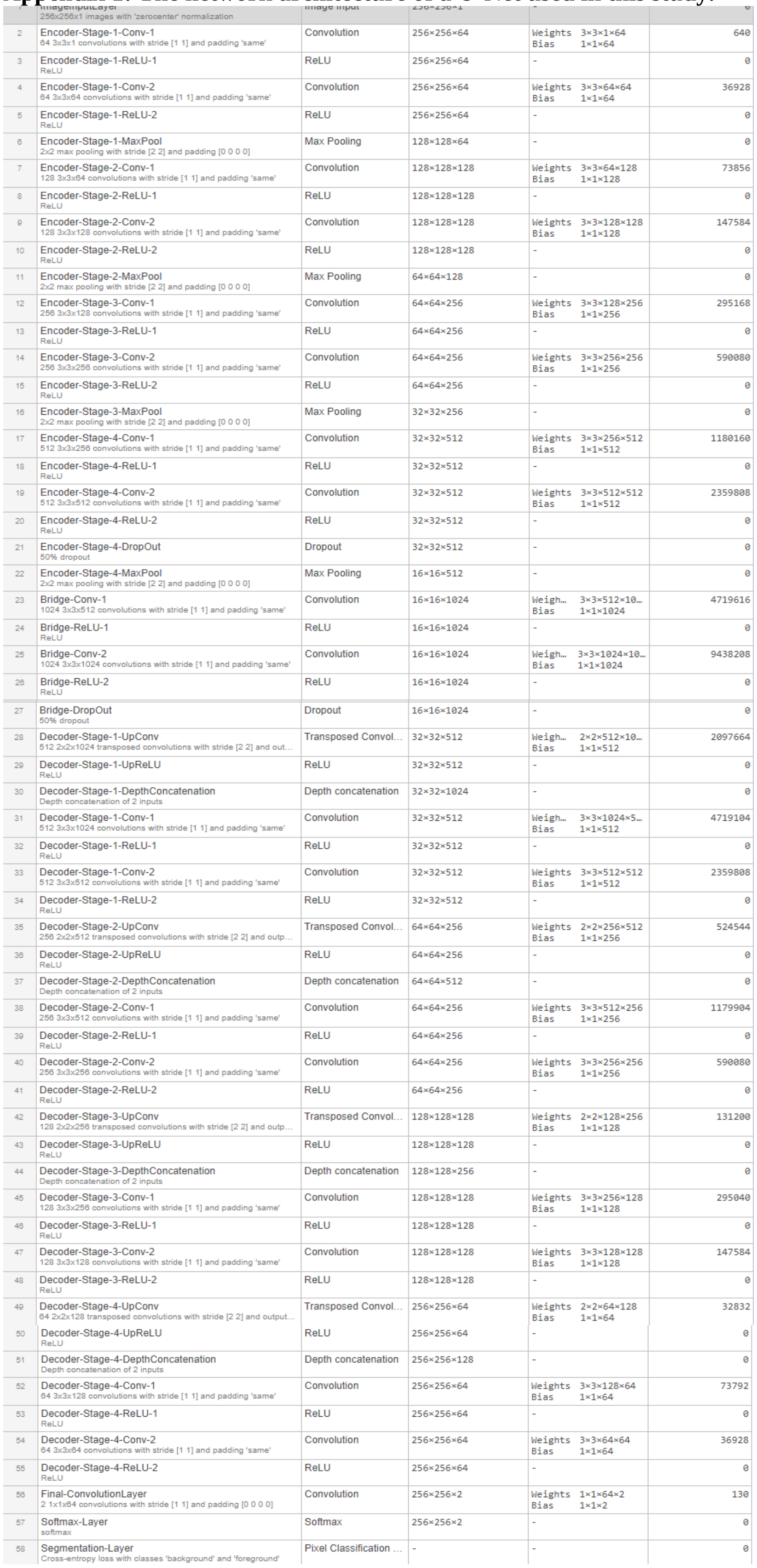
Appendix

The code used in this thesis can be found at:

https://github.com/MrMisnomer/USKneeCNN

This contains example networks and most of the code that can be transferred over for most Ultrasound segmentations.

Appendix 1: The network architecture of a U-Net used in this study.



1	ImageInputLayer 256x256x1 images with 'zerocenter' normalization	Image Input	256×256×1	-	0
2	Encoder-Stage-1-Conv-1 64 3x3x1 convolutions with stride [1 1] and padding 'same'	Convolution	256×256×64	Weights 3×3×1×64 Bias 1×1×64	640
3	Encoder-Stage-1-ReLU-1 ReLU	ReLU	256×256×64	-	0
4	Encoder-Stage-1-Conv-2 64 3x3x64 convolutions with stride [1 1] and padding 'same'	Convolution	256×256×64	Weights 3×3×64×64 Bias 1×1×64	36928
5	Encoder-Stage-1-ReLU-2 ReLU	ReLU	256×256×64	-	0
6	Encoder-Stage-1-MaxPool 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	128×128×64	-	0
7	Encoder-Stage-2-Conv-1 128 3x3x64 convolutions with stride [1 1] and padding 'same'	Convolution	128×128×128	Weights 3×3×64×128 Bias 1×1×128	73856
8	Encoder-Stage-2-ReLU-1 ReLU	ReLU	128×128×128	-	0
9	Encoder-Stage-2-Conv-2 128 3x3x128 convolutions with stride [1 1] and padding 'same'	Convolution	128×128×128	Weights 3×3×128×128 Bias 1×1×128	147584
10	Encoder-Stage-2-ReLU-2 ReLU	ReLU	128×128×128	-	0
11	Encoder-Stage-2-MaxPool 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	64×64×128	-	0
12	Encoder-Stage-3-Conv-1 256 3x3x128 convolutions with stride [1 1] and padding 'same'	Convolution	64×64×256	Weights 3×3×128×256 Bias 1×1×256	295168
13	Encoder-Stage-3-ReLU-1 ReLU	ReLU	64×64×256	-	0
14	Encoder-Stage-3-Conv-2 256 3x3x256 convolutions with stride [1 1] and padding 'same'	Convolution	64×64×256	Weights 3×3×256×256 Bias 1×1×256	590080
15	Encoder-Stage-3-ReLU-2 ReLU	ReLU	64×64×256	-	0
16	Encoder-Stage-3-MaxPool 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	32×32×256	-	0
17	Encoder-Stage-4-Conv-1 512 3x3x256 convolutions with stride [1 1] and padding 'same'	Convolution	32×32×512	Weights 3×3×256×512 Bias 1×1×512	1180160
18	Encoder-Stage-4-ReLU-1 ReLU	ReLU	32×32×512	-	0
19	Encoder-Stage-4-Conv-2 512 3x3x512 convolutions with stride [1 1] and padding 'same'	Convolution	32×32×512	Weights 3×3×512×512 Bias 1×1×512	2359808
20	Encoder-Stage-4-ReLU-2 ReLU	ReLU	32×32×512	-	0
21	Encoder-Stage-4-DropOut 50% dropout	Dropout	32×32×512	-	0
22	Encoder-Stage-4-MaxPool 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	Max Pooling	16×16×512	-	0
23	Bridge-Conv-1 1024 3x3x512 convolutions with stride [1 1] and padding 'same'	Convolution	16×16×1024	Weigh… 3×3×512×10… Bias 1×1×1024	4719616
24	Bridge-ReLU-1 ReLU	ReLU	16×16×1024	-	0
25	Bridge-Conv-2 1024 3x3x1024 convolutions with stride [1 1] and padding 'same'	Convolution	16×16×1024	Weigh… 3×3×1024×10… Bias 1×1×1024	9438208
26	Bridge-ReLU-2 ReLU	ReLU	16×16×1024	-	0
27	Bridge-DropOut 50% dropout	Dropout	16×16×1024	-	0
28	Decoder-Stage-1-UpConv 512 2x2x1024 transposed convolutions with stride [2 2] and out…	Transposed Convol…	32×32×512	Weigh… 2×2×512×10… Bias 1×1×512	2097664
29	Decoder-Stage-1-UpReLU ReLU	ReLU	32×32×512	-	0
30	Decoder-Stage-1-DepthConcatenation Depth concatenation of 2 inputs	Depth concatenation	32×32×1024	-	0
31	Decoder-Stage-1-Conv-1 512 3x3x1024 convolutions with stride [1 1] and padding 'same'	Convolution	32×32×512	Weigh… 3×3×1024×5… Bias 1×1×512	4719104
32	Decoder-Stage-1-ReLU-1 ReLU	ReLU	32×32×512	-	0
33	Decoder-Stage-1-Conv-2 512 3x3x512 convolutions with stride [1 1] and padding 'same'	Convolution	32×32×512	Weights 3×3×512×512 Bias 1×1×512	2359808
34	Decoder-Stage-1-ReLU-2 ReLU	ReLU	32×32×512	-	0
35	Decoder-Stage-2-UpConv 256 2x2x512 transposed convolutions with stride [2 2] and outp…	Transposed Convol…	64×64×256	Weights 2×2×256×512 Bias 1×1×256	524544
36	Decoder-Stage-2-UpReLU ReLU	ReLU	64×64×256	-	0
37	Decoder-Stage-2-DepthConcatenation Depth concatenation of 2 inputs	Depth concatenation	64×64×512	-	0
38	Decoder-Stage-2-Conv-1 256 3x3x512 convolutions with stride [1 1] and padding 'same'	Convolution	64×64×256	Weights 3×3×512×256 Bias 1×1×256	1179904
39	Decoder-Stage-2-ReLU-1 ReLU	ReLU	64×64×256	-	0
40	Decoder-Stage-2-Conv-2 256 3x3x256 convolutions with stride [1 1] and padding 'same'	Convolution	64×64×256	Weights 3×3×256×256 Bias 1×1×256	590080
41	Decoder-Stage-2-ReLU-2 ReLU	ReLU	64×64×256	-	0
42	Decoder-Stage-3-UpConv 128 2x2x256 transposed convolutions with stride [2 2] and outp…	Transposed Convol…	128×128×128	Weights 2×2×128×256 Bias 1×1×128	131200
43	Decoder-Stage-3-UpReLU ReLU	ReLU	128×128×128	-	0
44	Decoder-Stage-3-DepthConcatenation Depth concatenation of 2 inputs	Depth concatenation	128×128×256	-	0
45	Decoder-Stage-3-Conv-1 128 3x3x256 convolutions with stride [1 1] and padding 'same'	Convolution	128×128×128	Weights 3×3×256×128 Bias 1×1×128	295040
46	Decoder-Stage-3-ReLU-1 ReLU	ReLU	128×128×128	-	0
47	Decoder-Stage-3-Conv-2 128 3x3x128 convolutions with stride [1 1] and padding 'same'	Convolution	128×128×128	Weights 3×3×128×128 Bias 1×1×128	147584
48	Decoder-Stage-3-ReLU-2 ReLU	ReLU	128×128×128	-	0
49	Decoder-Stage-4-UpConv 64 2x2x128 transposed convolutions with stride [2 2] and output…	Transposed Convol…	256×256×64	Weights 2×2×64×128 Bias 1×1×64	32832
50	Decoder-Stage-4-UpReLU ReLU	ReLU	256×256×64	-	0
51	Decoder-Stage-4-DepthConcatenation Depth concatenation of 2 inputs	Depth concatenation	256×256×128	-	0
52	Decoder-Stage-4-Conv-1 64 3x3x128 convolutions with stride [1 1] and padding 'same'	Convolution	256×256×64	Weights 3×3×128×64 Bias 1×1×64	73792
53	Decoder-Stage-4-ReLU-1 ReLU	ReLU	256×256×64	-	0
54	Decoder-Stage-4-Conv-2 64 3x3x64 convolutions with stride [1 1] and padding 'same'	Convolution	256×256×64	Weights 3×3×64×64 Bias 1×1×64	36928
55	Decoder-Stage-4-ReLU-2 ReLU	ReLU	256×256×64	-	0
56	Final-ConvolutionLayer 2 1x1x64 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	256×256×2	Weights 1×1×64×2 Bias 1×1×2	130
57	Softmax-Layer softmax	Softmax	256×256×2	-	0
58	Segmentation-Layer Cross-entropy loss with classes 'background' and 'foreground'	Pixel Classification …	-	-	0