

Abstract

In the field of ultrasound imaging, it has been theorized that imaging noise, known as speckle, is the product of microscopic scatterers and abnormalities within the imaged tissue. This would result in certain speckling patterns revealing themselves over large datasets, which could be utilized to identify minuscule lesions within tissues, potentially creating a method to predict the early formation of tumors. Such a dataset would be difficult to analyze by hand, but machine learning algorithms could be used to recognize patterns in a effective manner. As of now, few attempts have been made to utilize machine learning in order to predict scatterer placement from ultrasound scans.

In order to initiate machine learning, first a computational simulation must be constructed to consistently and accurately reproduce experimental data. Using Field II, a MATLAB-based program for ultrasound modelling, simulations were created to replicate data produced from experimental phantoms made from glass beads and agarose gel. These simulations were designed to account for bead placement and size, as well as experimental conditions. Comparisons between simulations and experimental data using statistical analysis show that ultrasound images can accurately be predicted using computational methods. With these software programs, it becomes possible to train a machine learning algorithm to recognize speckling pattern, which may allow for the resolution of previously unresolvable scatterers.

Designing Simulated Radio Frequency Ultrasound Traces for the Training of Machine Learning Algorithms

Naomi Brandt

A thesis submitted to the Department of Physics in partial
fulfillment of the requirements for the degree of Bachelor of Arts
in Physics

Mount Holyoke College

May 2, 2019

Approved by

Maria Teresa Herd, Department of Physics, Mount Holyoke College

Kerstin Nordstrom, Department of Physics, Mount Holyoke College

Amy Camp, Department of Biology, Mount Holyoke College

Acknowledgements

First and foremost, I would like to thank my thesis advisor, Dr. Maria Teresa Herd, for inviting me into her lab, mentoring me, and helping me through every step of the thesis process. None of this would have been possible without her constant patience and help. I would also like to thank the other members of the Herd Lab Group, and in particular Nguyen Nguyen for all of her input in this research.

In addition, I am grateful for Mount Holyoke College and the Physics Department, not just for funding my research, but also for giving me four unbelievable years of excellent education and friends. I could not have done this without the nonstop support and guidance of all Mount Holyoke Physics Department faculty, staff, and students. I could not have asked for a better group of people to work, learn, and grow alongside.

Contents

Abstract	i
Acknowledgements	iii
List of Figures	vii
1 Introduction	1
1.1 Ultrasound RF and B-Mode Imaging	3
1.2 Acoustic Speckle	5
1.3 Computational Ultrasound	7
2 Theory	10
2.1 Gaussian Distribution of Speckle	10
3 Materials and Methods	13
3.1 Experimental Phantom Design	13
3.2 Experimental Ultrasound	14
3.3 Simulated Ultrasound	17
3.3.1 Field-II	17
3.3.2 Computational Phantom Design	18

3.4 Analysis	19
4 Results	25
4.1 Comparisons of Experimental and Simulated Traces	25
5 Discussion	31
5.1 Future Work with Machine Learning	35
References	39
Appendices	42
A MATLAB Program for Processing Experimental Data	42
B MATLAB Program for Designing Computational Phantom	44
C MATLAB Program for Performing Simulations	46
D MATLAB Program for Comparing Datasets	48
E Table of Overlaid and Offset Simulated Areas and Differences	51

List of Figures

1.1	An example of a prenatal b-mode ultrasound image of a human fetus.	5
1.2	A pair of images demonstrating the obscuring effect of speckle on ultrasound imaging.	6
3.1	A photograph of the experimental phantom with a ruler for scale.	13
3.2	A diagram of the ultrasound experimental setup.	22
3.3	A plot of response time versus response of experimental ultrasound data of a 1mm bead suspended in 0.7% agarose gel.	23
3.4	The averaged experimental RF ultrasound trace centered on a 1mm bead.	24
4.1	Stacked plots of an experimental ultrasound trace in relation to the depth of a 1mm glass bead in agarose gel and its associated simulated trace.	26
4.2	Comparisons between the averaged experimental and simulated responses centered on a 1mm glass bead.	28
4.3	Comparisons between offset averaged experimental and simulated responses of a 1mm glass bead.	28
4.4	A comparison between the averaged experimental and simulated data to demonstrate consistency between simulations.	29

5.1	A simulated ultrasound trace with attenuated speckle to demonstrate the "ringing" effect.	33
5.2	A diagram illustrating the procedural differences between classical machine learning and neural networks.	36

Chapter 1

Introduction

The concept of ultrasound and its use in identifying the location of objects was first proposed in the 1790s by Lazzaro Spallanzani, who observed the way in which bats moved while blindfolded. While he was correct in his theory that they used noises that were inaudible to the human ear to navigate, his findings were largely ignored until early 20th century. It was at this time, during the start of WWI, that the Allies began to investigate the use of these high-frequency sounds as a source of detection for enemy submarines. It wasn't until 1942 that ultrasound was first used in medical imaging by Karl Dussik in an attempt to identify brain tumors [1].

Following Dussik's work, the medical community began to see an resurgence in applied ultrasound in a clinical setting. Throughout the 1940s and 1950s, other scientists experimented to find the best methods for ultrasound imaging. These techniques began with submerging the subject in a water bath, with the ultrasound transducer situated some distance away in the water, but soon moved towards having transducer to skin contact, with oils and gels as the intermediary between the surfaces. By 1960, the first contact scanners

were made available for purchase, and ultrasound soon replaced the more hazardous x-rays as the primary prenatal imaging tool. Since then, researchers have continued to optimize ultrasound equipment and processing, moving towards small, light, transducers and screens, as well as increasing the resolution and diagnostic capabilities of ultrasound imaging. [2]

Currently, ultrasound has dozens of use in modern medicine, from imaging blood flow for potential clots or aneurysms, to diagnosing and breaking up kidney stones. Ultrasound fills an important niche in clinical settings as a versatile, safe, yet relatively inexpensive, imaging technology, making diagnostic imaging readily available to the majority of the population. However, there are still some needs that remain unmet by ultrasound imaging, and the scientific community continues to develop new methods to fill these gaps [2].

One emerging focus in the field of ultrasound imaging is its potential applications in the early diagnosis and treatment of various cancers. While ultrasound has historically held a place in the identification of cancerous tumors, it functions best when visualizing tumors past a certain size, which is generally limited by the wavelength of the emitted pulse (described further in Section 1.1). In addition, small structures in ultrasound images tend to be obscured by a grainy interference pattern across the image. This presents a significant flaw in ultrasound as a diagnostic tool, as cancer treatments are most successful at the earlier stages of tumorigenesis [2, 3].

However, the aforementioned interference patterns, while often treated as troublesome noise, may hold the key to resolving microscopic scatterers within tissue. This patterning, known as speckle, is due to unresolved scatterers in an imaged tissue, and will be discussed in more depth in Section 1.2. It is theorized that, since speckle holds structural information on the size and position of microscopic scatterers, a machine learning algorithm could be

used to identify features within speckling patterns and predict the location of said scatterers.

The Herd Lab group has been working with this hypothesis for some time, and in 2015, Huma Yusuf produced a thesis investigating the theory behind speckle and its proposed models [4]. In 2016, Colbie Chinowsky continued Yusuf's work by performing experimental ultrasound scans and building the foundation to create accurate simulations of experimental phantoms [5].

The purpose of this study will be to take the next logical step in the research presented by Yusuf and Chinowsky. This thesis will provide an introduction to ultrasound imaging, speckle, and computational ultrasound. It will then identify a potential method of locating unresolved scatterers in ultrasound images using speckle, by training a machine learning algorithm on simulated ultrasound traces. Finally, the thesis will discuss the experimental methods, results, and conclusions in designing the simulations, before ending with a analysis of available machine learning algorithms.

1.1 Ultrasound RF and B-Mode Imaging

In ultrasound imaging using pulser-reciever transducers, a pulsed wave-packet is released from the transducer and hits the tissue, causing the waves to scatter and reflect back. The amplitudes of these reflections are primarily dictated by the amplitude reflection factor, R , which is the ratio of the reflected wave relative to the incident wave and can be found using:

$$R = \frac{Z_2 - Z_1}{Z_2 + Z_1} \quad (1.1)$$

Where Z_1 and Z_2 are the characteristic acoustic impedances of the two media, described

by

$$Z = \rho c \tag{1.2}$$

With ρ denoting the density of the medium, and c denoting the speed of sound within the medium. As such, the amplitude of the reflected wave is largely dependent on characteristics of both media. This allows for a wide variety of biological tissues to be identified by ultrasound, as the amplitude reflection factor varies from tissue to tissue [2].

Once the wavepackets are attenuated and reflected at the interface of the media, they are sent back to hit the transducer, causing the piezoelectric within the transducer to vibrate. The piezoelectric translates the amplitude of these vibrations into voltages, which is recorded and processed. The non-processed version of each voltage trace is called a radio-frequency (RF) trace [2].

Brightness mode, or b-mode, is a form of ultrasound imaging, commonly used for the visualization of fetuses or internal organs. B-mode images are characterized as two dimensional figures, typically in gray-scale, which display a cross-section of the targeted tissue [Fig. 1.1]. Since the pulsed wave packets are sent out in discrete lines, the two-dimensionality of B-mode images is achieved by concatenating many RF signals together. The voltages of the signals are then weighted on a logarithmic scale, before being assigned brightness values, with higher amplitudes correlating to brighter points [2].

While b-mode imaging produces a fairly accurate representation of larger tissues and organs, small scatterers are frequently non-resolvable. In general, there are two sources of this limitation. The first is speckle, which will be discussed in Section 1.2. The second is the

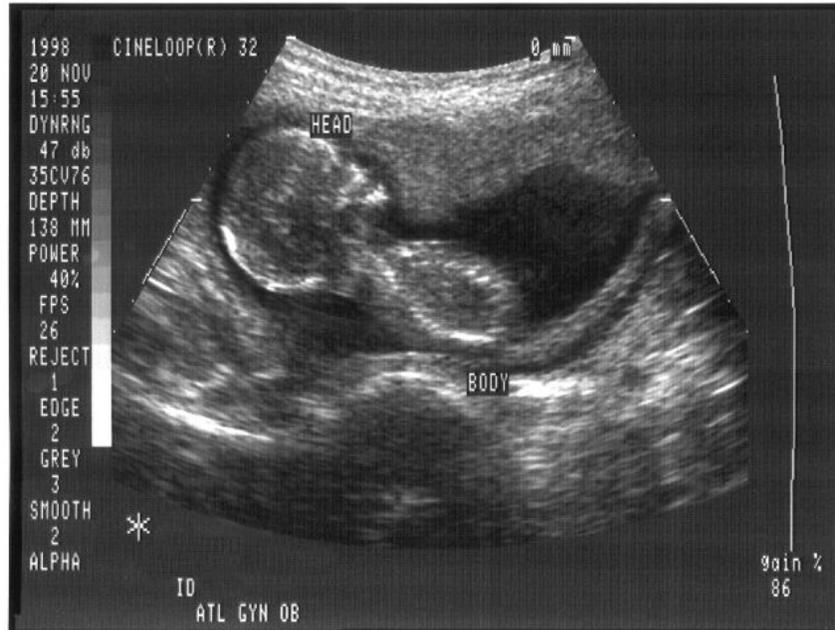


Figure 1.1: An example of a prenatal b-mode ultrasound image of a human fetus. The scattering of ultrasound is denoted by shades of gray, with the highest intensity response shown as white and the lowest as black [6].

pulse length of the ultrasound wave. This pulse length is defined as the product of the number of cycles within the wave packet and its associated wavelength. Transducers producing high frequency wavepackets, and therefore short pulse lengths, tend to result in higher resolution images. However, a balance must be struck, as while high frequency transducers may resolve scatterers more accurately, they are attenuated quickly in soft tissue, resulting in a short distance of penetration [3]. As such, much work has been done towards perfecting this balance, or searching for new methods to resolve microscale scatterers [7].

1.2 Acoustic Speckle

Acoustic speckle, the sound-based counterpart to laser speckle, refers to a certain form of granule patterning that occurs on b-mode ultrasound images [Fig. 1.2] [8, 9]. In imaging

science, speckle is often treated as interference similar to electrical noise, resulting in researchers pursuing the development of algorithms to minimize its effects [9]. However, while this approach towards speckle mimics that of electrical noise, speckle’s characteristics proves the two to be incomparable.

Electrical noise is generally considered to be a random baseline signal, which has little to no structural meaning and only results in obfuscating low-amplitude signals. The randomness of noise means that multiple datasets of the same experiment taken under the same conditions will exhibit differences in its baseline [8]. Ultrasound, on the other hand, demonstrates similar speckling behavior throughout various scans. This suggests that while speckle may have certain stochastic aspects, it is not purely randomized noise. Instead, it has been theorized to be the result of constructive and destructive scattering patterns due to structural differences and unresolvable scatters in the imaged tissue [8, 9].

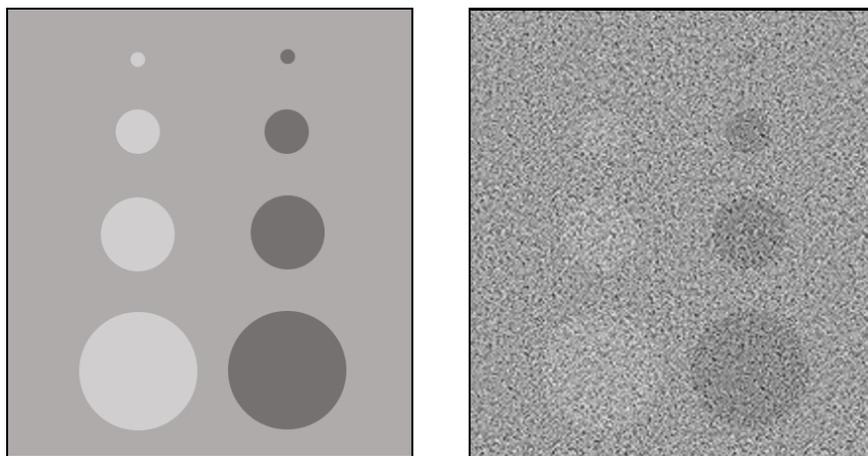


Figure 1.2: A pair of images demonstrating the obscuring effect of speckle on ultrasound imaging. A theoretical tissue sample is shown on the left, with a series of high and low scattering areas. This same tissue is replicated on the right, with an overlaid speckling pattern.

This distinction, while seemingly minute, forms the basis for a novel approach to the

identification of small scatterers. Since speckle is not random, speckling patterns include information on the position and density of scatterers that may not be resolvable by usual ultrasound imaging methods. As such, it is reasonable to look towards using these patterns as an alternative method of identifying microscale structures.

While this approach towards counteracting the limitations of ultrasound resolution may be somewhat recent, it is not unprecedented. In 1995, Chen et. al. [10] demonstrated speckle’s potential for tissue analysis by using it to track the movement of tissue across a number of ultrasound scans. They showed that speckle can reliably be used to track tissue movement across a variety of tissue samples, proving that the information stored in speckling patterns can be used to identify structural changes and movement. More recently, in 2015, Aalamifal et. al. [11] used speckle to classify liver and kidney tissue samples. The group found that this method allows for high accuracy classification, further indicating the potential of speckle in identifying tissue differences that may not be observable in b-mode imaging.

1.3 Computational Ultrasound

Abbott and Thurstone (1979) and later Wagner et. al. (1983) performed derivations using Rayleigh distributions to model speckle as high order probability distributions. However, as described later in Section 2.1, stochastic models of speckle become increasingly complex as they reach higher order statistics, to the point at which it becomes impractical to solve such equations by hand [9, 12].

Therefore, it is appropriate to use computational methods in order to fully model ultra-

sound scattering and speckle. A variety of programs have been developed to create simulated ultrasound data. These algorithms are able to efficiently solve high order statistical functions given the inputs of the experimental setup and phantom, allowing them to produce accurate models of ultrasound data.

In 1983, only five years after the term "acoustic speckle," was coined, a research team led by D. R. Foster developed one of the first programs to computationally simulate speckle. Their model was designed to take the three dimensional geometry of both the phantom and the transducer into account, a method that is now commonplace in ultrasound simulations. Similarly, they designed microscopic scattering structures within computational phantom as an assortment of randomly placed point scatterers, which is similar to our phantom design discussed later, in Section 3.3.2. Although other aspects of the model were a simplified version of what can be achieved today, the group demonstrated the capability of accurately simulating ultrasound traces with speckle [13].

In general, many of the more recent algorithms have been used to remove speckling patterns from experimental data, in order to achieve higher resolution b-mode images. However, with the advancements of machine learning and neural networks, it has recently become much more feasible to use computational methods to draw structural information from speckling patterns. Machine learning programs have proved to be capable of pattern recognition, often finding regularities in datasets that are overlooked by researchers. In terms of ultrasound, machine learning has the potential to identify indiscernible patterns within speckle, and use these patterns to pinpoint unresolved scatterers.

The research performed by Aalamifal et. al. [11] represents a successful attempt at using machine learning and speckle for the classification of ultrasound data, establishing

the capability of such methods in ultrasound. However, the research group had access to the large experimental datasets necessary for training a machine learning algorithm. For the purposes of this study, such datasets are inaccessible, further necessitating the use of simulated ultrasound.

If b-mode ultrasound scans can be consistently and accurately simulated using computational methods, it would be possible to create large sets of ultrasound data without using the resources to experimentally produce the same amount of data. Therefore, provided that the simulations are precise, computational ultrasound could be used to train machine learning algorithms as efficiently as experimental ultrasound. As such, this research will investigate the potential of using machine learning to recognize speckling patterns in simulated ultrasound data, in order to locate scatterers that are unresolvable by conventional methods.

Chapter 2

Theory

2.1 Gaussian Distribution of Speckle

In 2008, Daba et. al. [14] identified two forms of speckling behavior in ultrasound, known as developed and partially developed speckle. This distinction is made depending on the number of scatterers within the focus of the ultrasound beam. When there are a large number of structural differences that cause scattering within the range of the focus, the speckling patterns are developed, and generally can be modeled as a Gaussian distribution of amplitudes across the ultrasound image. However, more complex stochastic methods of modeling are needed when there are fewer than 20 scatterers, resulting in partially developed scattering.

For the purposes of this research, and the resources at our disposal, we will assume fully developed scattering in both imaged and simulated ultrasound. This assumption is appropriate for our purposes, as the simulations were set to have significantly more than 20 general scatterers (on the order of 10^6 scatterers), and the agarose-based experimental phantoms

naturally have large numbers of unresolvable scatterers, due to the protein structure within the gel [14].

As such, it becomes reasonably straightforward to derive the first order probability density function for the intensity of speckle, as demonstrated by Wagner et. al. [12]. Gaussian distribution functions take the general form of

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-a)^2}{2\sigma^2}\right\} \quad (2.1)$$

in which σ is the standard deviation and a is the mean of the function. For this to be used to describe a wave, which can be described by Euler's equation to have both real and imaginary terms, the probability distribution can be defined as the product of two Gaussian functions, one for each term. In the following equation, we will let ϕ_r represent the real valued components and ϕ_i represent the imaginary components. Additionally, in this case, the mean can be set to zero, since positive and negative intensities are equally likely.

$$P(\phi_r, \phi_i) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{(\phi_r^2 + \phi_i^2)}{2\sigma^2}\right\} \quad (2.2)$$

Equation 2.2 is known as a circular Gaussian probability distribution function, and is demonstrated in terms of the two components. However, due to the nature of b-mode imaging in ultrasound, it becomes more useful to consider this equation in terms of amplitude, V , in which $V = (\phi_r^2 + \phi_i^2)^{1/2}$, giving us

$$P(V) = \frac{V}{\sigma^2} \exp\left\{-\frac{V^2}{2\sigma^2}\right\} \quad (2.3)$$

Since negative probability is impossible, this equation is then reformed into a piece-wise function such that

$$P(V) = \begin{cases} \frac{V}{\sigma^2} \exp\left\{-\frac{V^2}{2\sigma^2}\right\}, & V \geq 0 \\ 0, & V < 0 \end{cases} \quad (2.4)$$

This equation takes the form of a Rayleigh distribution function, and demonstrates the basic probability distribution of speckle.

It should be noted that this derivation followed only the first order probability distribution, and therefore is limited as a model. Higher order statistics can be applied to define speckle in a more realistic way. However, derivations of these probability distributions requires significantly more complex calculations. This serves to further illustrate the necessity of using rigorous computational methods, both to calculate complex scattering patterns, and to identify overlooked patterning with machine learning.

Chapter 3

Materials and Methods

3.1 Experimental Phantom Design



Figure 3.1: A photograph of the experimental phantom with a ruler for scale. The bead to the left is 0.5mm in diameter, the middle is 1.3mm, and the bead to the right is between 2.9mm.

Experimental phantoms were created at room temperature using 0.7% agarose gel, glass beads of varying sizes, and petri dishes. The petri dishes were 35mm in diameter and 10 mm in height. Agarose gel was microwaved until it became liquid, and then was used to fill the dishes about halfway (about 5mm). After allowing the gel to cool for about 2 minutes, beads were placed in a line along the diameter of the dish, with one bead of 0.5mm in diameter, one bead with diameter 1.3mm, and one bead between 2.9mm (Fig. 3.1). Each bead was placed with approximately 10mm of spacing between their centers, with the centers of the 0.5mm and 2.9mm each 7.5mm away from the edge of the dish (Fig. 3.1). Agarose gel was placed over the beads, filling the petri dish with 8mm total of gel. The phantom was allowed to cool and harden for 5 hours before use.

3.2 Experimental Ultrasound

Pulse-echo ultrasounds were performed using an Olympus 5 MHz center frequency, single element, immersion transducer with a focal length of 52.172mm. A series of three stepping motors, one for each axis, were used in conjunction with Computer Optimized Stepper Motor Operating System (COSMOS) software to accurately move the transducer (Fig. 3.2). Both the phantom and transducer were lowered into a 23°C bath of 18 Mohm water, and the transducer was connected to a Olympus 5073 Pulser/Receiver, which was connected to a Tektronix TDS 3014C Digital Phosphor Oscilloscope (Fig. 3.2). The phantom was oriented such that the largest scatterer (2.9mm) was at the top and the smallest (0.5mm) was at the bottom.

The transducer was intended to be positioned at 49.672mm away from the phantom, to

account for the focal length and the depth of the beads (Fig. 3.2). This distance was determined by setting the focus (52.172mm) of the scope to the back surface of the phantom and subtracting the depth of the beads (2.5mm). However, a mistake in the experimental setup resulted in the focus being set at the front of the phantom instead, placing the front surface 54.172mm away from the transducer. This likely resulted in some accidental amplification of the phantom surface, but should not have severely affected the rest of the results.

The center of the oscilloscope was set at the time corresponding with this distance, at 73.3 μ s. Times were found using the motion equation:

$$x = \frac{1}{2}vt \tag{3.1}$$

which can be rewritten as

$$d = \frac{1}{2}c_s t \tag{3.2}$$

Where d is the distance between the transducer, c_s is the speed of sound in water, and t is time. Solving for t results in the approximate time for the pulse to reach the front surface of the phantom and return to the receiver.

The transducer was moved across the width and height of the phantom, which was used to both center the transducer with the phantom, and to determine any possible misalignment between the face of the transducer and the surface of the phantom. By observing time changes of the echo from surface scattering of the phantom, we determined that there was a permanent gradient between 0.7-1.4mm horizontally and a gradient between 2.2-2.9mm vertically. These variations were due to imperfections in the phantom surface, likely from

human error while pouring the agarose gel into the dish, resulting in a slight curvature of the phantom surface. While these gradients were not ideal, they were not solvable by realigning the transducer or phantom, and therefore remained.

To center the transducer relative to the phantom, it was moved horizontally until the scattering due to the phantom's surface began to attenuate on the oscilloscope, which is indicative of the edges of the phantom. The stepping motor positions at each edge were recorded from COSMOS. The mean of these values was taken, and the stepping motor was placed at the central position. This procedure was repeated to find the center vertically.

Once the transducer was centered, data was taken from the oscilloscope with a vertical scale of 200mV and a horizontal scale of 4.0 μ s centered at 73.3 μ s, in order to obtain adequate visuals of both surface scattering and the 1.3mm bead scattering (Fig. 3.3). It was taken again at the same position with a horizontal scale of 1.0 μ s and a center at 77.45 μ s for a better visual of the bead scattering. Seven more scans were taken, each moving vertically down from the center at steps matching the size of the focal width. Focal width was found using the equation for diffraction limited angular resolution:

$$\theta = \frac{1.22\lambda}{D} \quad (3.3)$$

Where θ is the angular resolution, λ is wavelength, and D is the diameter of the transducer aperture. By substituting in the identities $\theta = \frac{w}{d}$, with w as focal width and d as focal distance, and $\lambda = \frac{c_s}{f}$, where f is frequency, the following equation can be derived:

$$w = \frac{1.22c_s}{Df}d \quad (3.4)$$

Using 1491m/s for the speed of sound in water, 0.0127m (0.5in) for the diameter of the transducer, 5×10^6 Hz for the frequency, and the previously given .052172m focal distance, it was found that the focal width was about 1.4mm. Therefore, the transducer was moved 1.4mm vertically for each scan.

The transducer was then centered once more, and 8 scans were taken as it was moved vertically up, each at the same step size as before. The centering of the scans was changed to account for the gradient of the surface, in order to keep the surface scattering at approximately the same position relative to the oscilloscope data. The center was moved to 76.5 μ s, then 75.97 μ s. One final set of data was taken, since the largest bead (2.9mm) had not been visible, likely because the scatterers were not aligned perfectly vertically. The transducer was moved until this scatterer was found, and data was taken.

3.3 Simulated Ultrasound

3.3.1 Field-II

Field-II is a MATLAB-based software package first developed in 1996 by Jørgen Arendt Jensen [15, 16]. This package uses spacial-impulse response modeling, as rigorously derived by Colbie Chinowsky [5], to replicate the response signal for a variety of ultrasound transducer geometries and apodization, or differential vibration intensities across the face of the transducer [17].

This range of geometries and apodization is achieved by simulating the transducer surface as a summation of small rectangles, each with their own average response signal. By break-

ing up the surface in such a way, it's possible to reasonably approximate not only square transducers, but triangular and circular ones as well. Since each section has an associated response, apodization patterns can also be mimicked, as the rectangles towards the edges can receive lower signals than those in the center [18].

These approximations are most accurate when the size of the rectangles is much smaller than the distance between the transducer and the focal point, which occurs when

$$l \gg \frac{w^2}{4\lambda} \quad (3.5)$$

where l is the focal distance, w is the length of the longest side of the rectangle, and λ is the associated wavelength [18]. This places the focal point in the far-field ultrasound range, which is appropriate for evaluating tissue differences in medical ultrasound.

3.3.2 Computational Phantom Design

In Field II, computation phantoms were constructed in a separate file and called upon by the main simulating program (see Appendices B and C). The phantoms were designed by inputting the approximate dimensions of the experimental phantom. It should be noted that the experimental phantoms were roughly cylindrical, while the computational phantoms were rectangular prisms. This should have no effect on the similarity of the traces, as no experimental traces were taken at the edge of the phantom.

After inputting the dimensions, 100,000 small, general scatterers were uniformly randomly placed within the confines of the phantom. The scatterers' amplitudes were generated using MATLAB's `randn` function, which creates arrays of values that fall within the

standard normal distribution, as given by the probability distribution function:

$$P(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2} \quad (3.6)$$

The scattering patterns due to these small scatterers can be used to mimic the expected patterns of speckle [19]. The size and placement of the glass beads were then input into the phantom program, assuming the beads were perfectly spherical. Due to the homogenous makeup of the bead, it was modeled by setting the amplitudes of the randomized scatterers that fell within its perimeter to a constant value. The general scatters, alternatively, were amplified by a variable amount, and thus maintained their Gaussian distribution of amplitudes. The values for both the bead amplitudes and the speckle amplitudes were found by comparing the responses from the experimental data to those from the simulations. As such, the bead amplitudes were set to a value of $2.945 \cdot 10^{22}$ and general scatterers were amplified by a factor of $0.4 \cdot 10^{22}$, values which were found to give the most similar results.

For the focus of this research, scattering due to the water/phantom interface was not a central concern. As such, this scattering was modeled by a highly-scattering point, which had an amplitude of $75 \cdot 10^{22}$, to match the amplitudes of the similar region in the experimental data.

3.4 Analysis

Comparisons between experimental and simulated ultrasound traces were performed using both qualitative and quantitative techniques. Qualitatively, the simulated and experimental data were observed alongside one another, to observe if the basic characteristics of the exper-

imental trace were appearing within the simulated version. These characteristics included bead position, size, and scattering amplitudes, as well as the general amplitude variations due to speckle (Fig 3.3).

In order to perform point-by-point quantitative comparisons, both datasets were required to be the same length. However, the experimental sets contained 10,000 data points, due to the sampling rate of the oscilloscope, whereas the simulated sets, by nature of the Field II program, tended to be on the order of around 2,000 points. Therefore, the squared value of dataset was cut to have the exact same start and endpoints, starting at 0.0540m and ending at 0.0615m. The datasets were then sorted into 500 bins with identical edges, with the contents of each bin averaged, resulting in the experimental and simulated datasets matching for every point.

After binning and averaging each trace, quantitative analysis was achieved using a variety of methods. In general, the comparisons were centered on the approximate position of the bead, to ensure that both the scattering amplitude due to the bead and the position and size of the bead were similar. The approximate placement and width of the bead was defined by estimating the starting and ending depths of the amplified scattering due to the bead. For the experimental trace, these points were defined at $X=56.42\text{mm}$ to $X=56.87\text{mm}$, demonstrating a width of 0.45mm. It should be noted this width is significantly smaller than the given 1mm diameter of the bead. This discrepancy is likely due to errors in the centering of the transducer in relation to the bead, causing it to only display the edge of the bead. The simulation was changed accordingly to represent the smaller width (Fig. 3.4).

To observe the potential accuracy of the simulations, 20 simulated traces were produced and overlaid with the experimental trace, and the area below each curve between 56.4mm

and 56.85mm was calculated. The simulations were then performed 20 more times, with the beads offset by one another by 0.59mm. In both cases, the differences between the experimental and simulated areas were taken and averaged across each sample set, as given in Section 4.1.

In addition to measuring the potential precision of the simulated traces, the consistent accuracy was also taken into account. As mentioned before in Section 3.3.2, the simulated phantoms were created with a randomized set of general scatterers, and as such demonstrated a level of randomness in the scattering due to the glass beads. To understand how consistently the simulations were replicating the experimental data, ten simulations of the same experimental setup were executed and overlaid with one another. The range of depths in which the peaks occurred from the bead was then calculated to determine the consistency of the simulated bead width and placement.

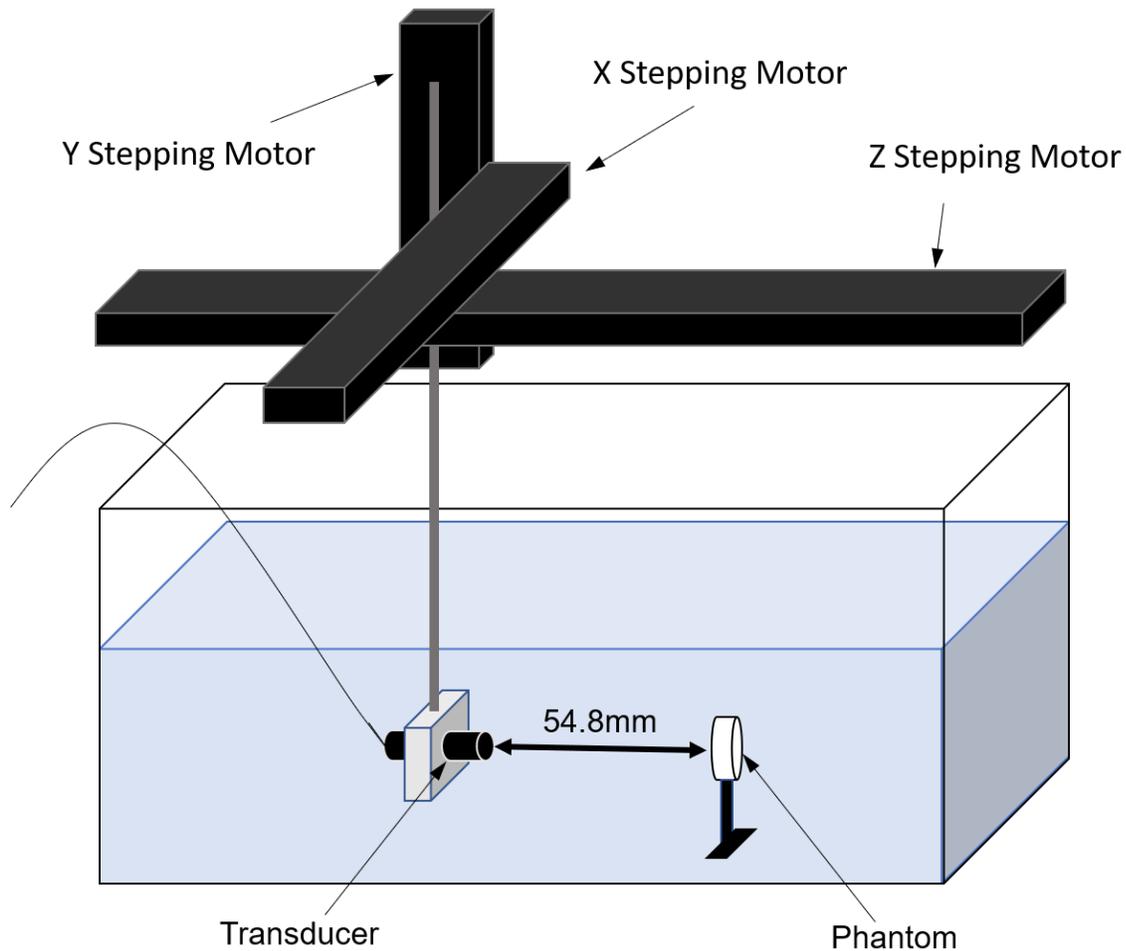


Figure 3.2: A diagram of the ultrasound experimental setup. The transducer and phantom have been labeled, and can be viewed as 54.8mm apart within the 18 M Ω water bath. The wire leaving the transducer is visible, and would be attached to an Olympus 5073 Pulser/Receiver, which is not in frame. The stepping motors have been labeled with their respective axes, as defined in the simulations.

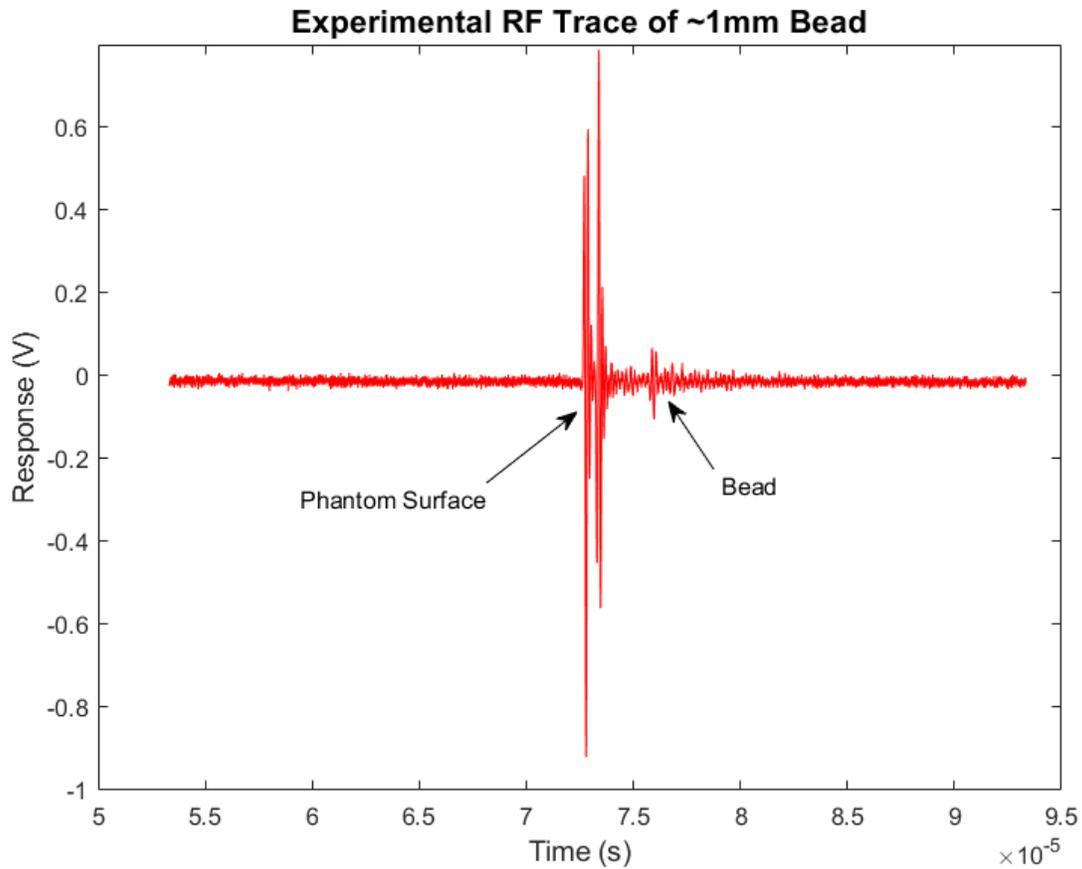


Figure 3.3: A plot of response time versus response of experimental ultrasound data of a 1mm bead suspended in 0.7% agarose gel. The response is measured in arbitrary units (a.u.). Scattering due to the surface of the phantom and due to the bead have been labeled. The small variations in the amplitude of the trace are due to ultrasound scattering off of microscopic structure within the agarose gel (speckle).

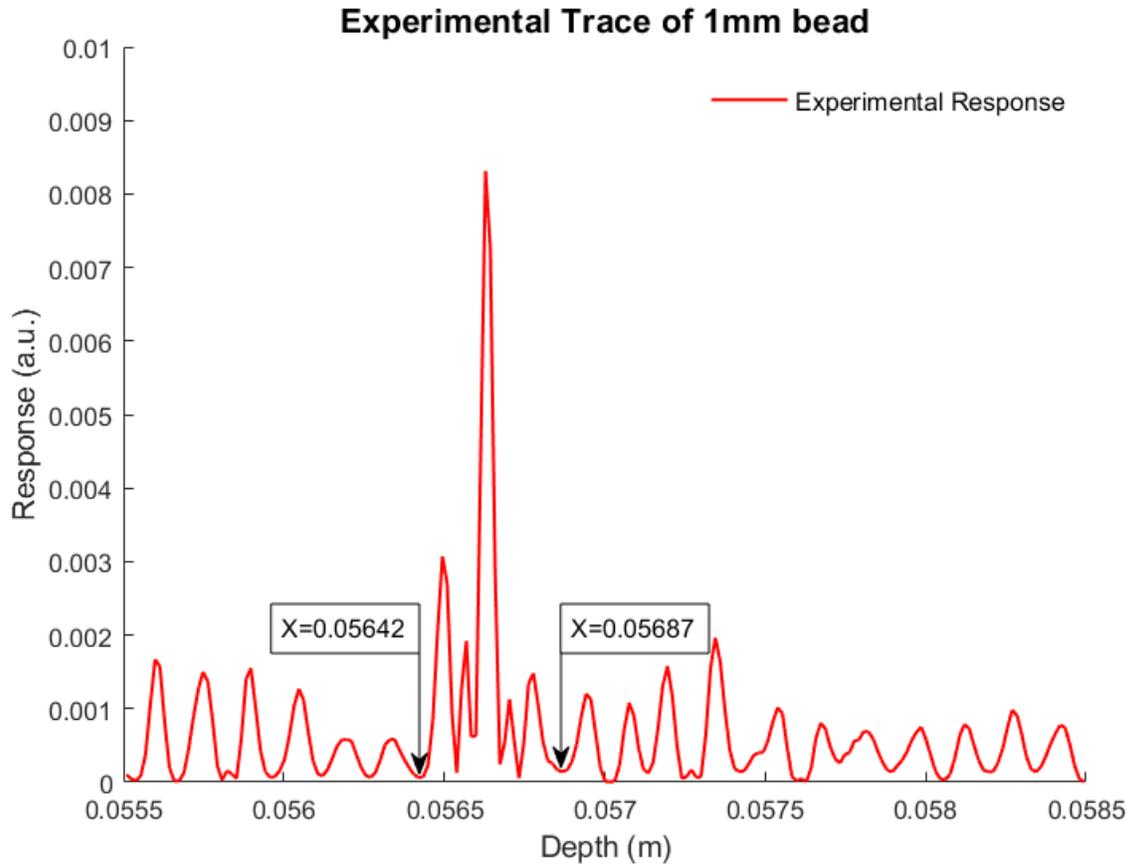


Figure 3.4: The averaged experimental RF ultrasound trace centered on a 1mm bead. The marked X values ($X=0.05642\text{m}$ (56.42mm), $X=0.05687\text{m}$ (56.78mm)) indicate the approximate starting and ending depth of the bead.

Chapter 4

Results

4.1 Comparisons of Experimental and Simulated Traces

Experimental traces of a 1mm glass bead in 0.7% agarose gel demonstrated the expected scattering patterns. As seen in Fig. 3.3, large peaks of 0.8a.u. occurred at approximately 72 μ s, corresponding to 54mm, which were indicative of scattering occurring due to the ultrasound's wavepackets first interacting with the gel surface. Later, at about 75 μ s (56mm), smaller peaks ranging from 0.05-0.1a.u. were observed, marking the placement of the glass bead (Fig. 3.3, 4.1).

Qualitatively, the two datasets generally demonstrated similar characteristics. The scattering due to the surface started at approximately the same point, 0.054m, with amplitudes reaching about ± 8 a.u., though the scattering patterns themselves were disparate, for reasons mentioned in Section 3.3.2. Scattering due to the 1mm glass bead regularly began at the same depth with approximately the same bead radius (Fig 4.1).

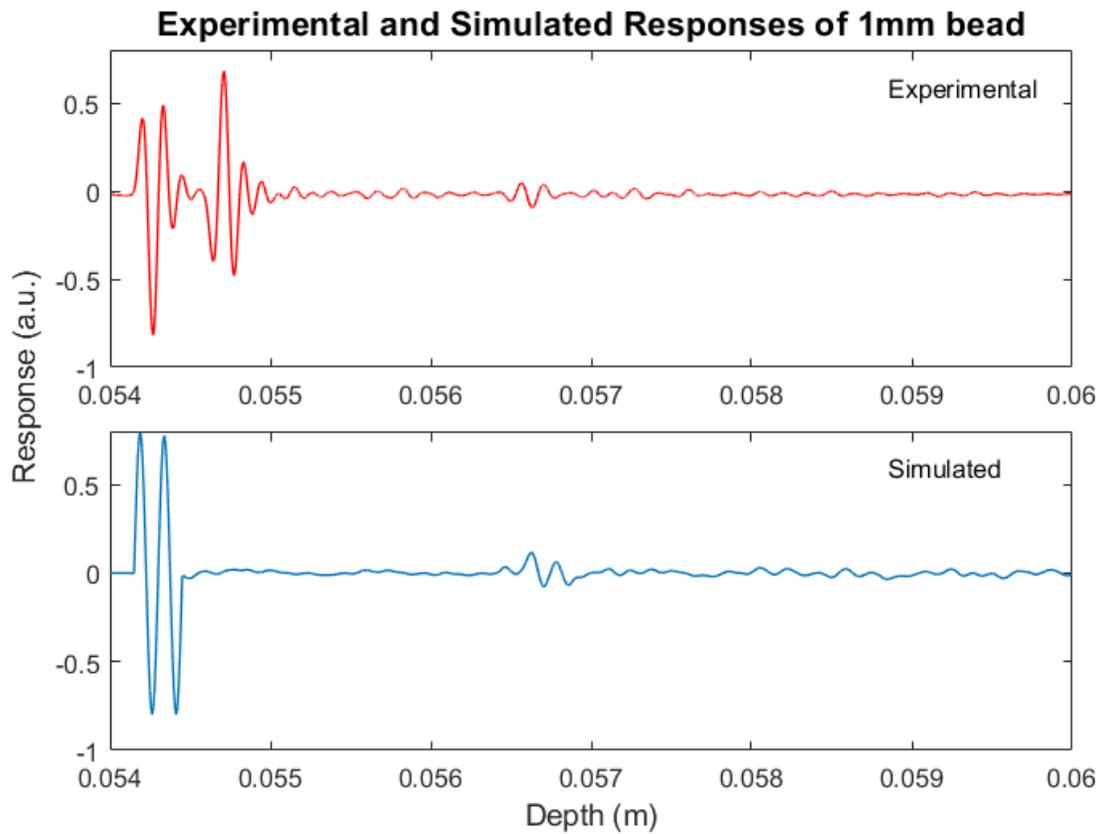


Figure 4.1: Stacked plots of an experimental ultrasound trace in relation to the depth of a 1mm glass bead in agarose gel (top) and its associated simulated trace (bottom). Scattering due to the 1mm bead can be seen at about 0.0565m in depth for both.

	Overlaid (n=20)	Offset (n=20)
Average Simulated Area	$4.2*10^{-7}$	$9.1*10^{-8}$
Simulated Area Standard Deviation	$1.2*10^{-7}$	$4.6*10^{-8}$
Average Area Difference	$2.7*10^{-7}$	$6.0*10^{-7}$

Table 4.1: A table containing the average areas and standard deviation under overlaid and offset simulated traces centered between 56.4mm and 56.85mm. The area differences were taken with respect to the area under the experimental curve ($6.9*10^{-7}$ a.u.*m).

These qualitative observations proved to stand up to quantitative comparisons. In directly comparing a single simulations centered on the bead’s scattering to the experimental data, the simulated peaks due to the bead largely matched the experimental trace in both location and amplitude (Fig. 4.2). In addition, the area below experimental trace was approximately $6.9*10^{-7}$ a.u.*m. When the simulated bead was placed at the same location as the experimental bead, as shown in Fig.4.2, the simulation exhibited an average area of $4.2*10^{-7}$ a.u.*m, resulting in an area difference of $2.71*10^{-7}$ a.u.*m, with a standard deviation of $1.2*10^{-7}$ a.u.*m. In comparison, when the beads were offset by 0.59mm from one another, the area under the curve averaged at $9.1*10^{-8}$ a.u.*m with a standard deviation of $4.6*10^{-8}$ a.u.*m, corresponding to a difference in area of $6.0*10^{-7}$ a.u.*m (Table 4.1).

In comparing the range of bead placements and responses across multiple simulations of the same experimental phantom, it was found that scattering due to the glass bead varied in both position and amplitude. The simulated peaks tended to match in amplitude to the experimental data, with some variation from simulation to simulation. The central positioning of the glass bead remained constant across all simulations, though the exact width of the scattering patterns differed slightly. It is worth noting, as well, that the width

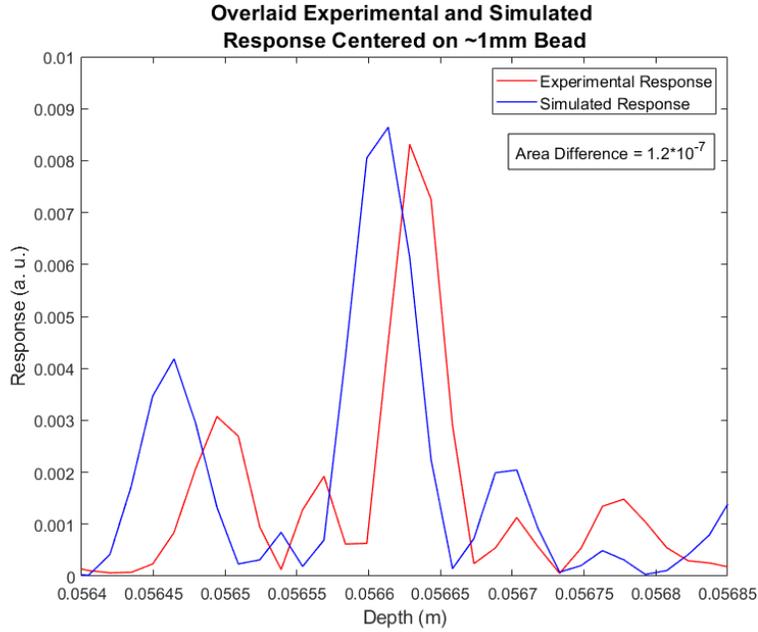


Figure 4.2: Comparisons between the averaged experimental and simulated responses centered on a 1mm glass bead. The plot includes the experimental response (red) overlaid with a simulated response (blue). The difference in area under the curves is labeled.

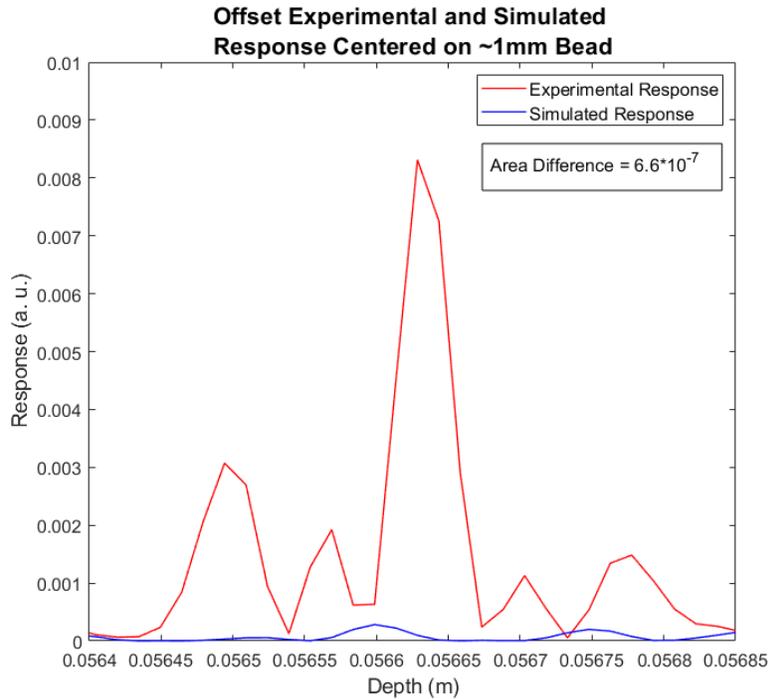


Figure 4.3: Comparisons between offset averaged experimental and simulated responses of a 1mm glass bead. The plot includes the experimental response (red) with a simulated response (blue) offset by 0.59mm. The average difference in area under the two responses is labeled.

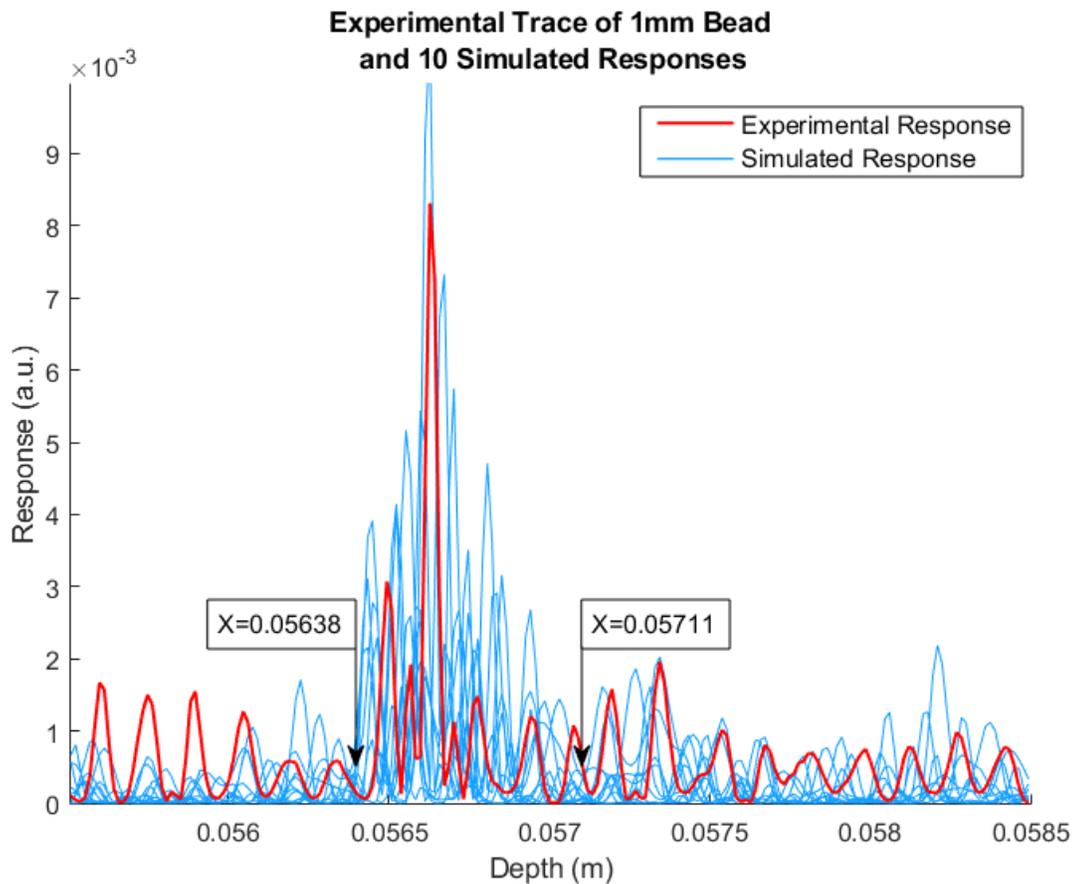


Figure 4.4: A comparison between the averaged experimental and simulated data to demonstrate consistency between simulations. The plot contains an overlay of 10 simulated responses (blue) using the same inputs and conditions. The experimental response (red) of the same setup is superimposed on top of the simulated responses. The marked X values ($X=0.05638$, $X=0.05711$) indicate the approximate starting and ending depth to the simulated beads.

of the echo from the simulated bead, though defined in the simulation as 0.45mm, tended to be larger by about 0.2-0.3mm. As such, over many simulations, the approximate width of the simulated bead was observed to be 0.73mm, which was 0.28mm larger than the experimental results (Fig. 4.4).

Chapter 5

Discussion

In comparing experimental RF ultrasound traces with their associated simulations, it proved possible to demonstrate that simulations performed using Field-II have the potential to be both qualitatively and quantitatively precise. The simulated phantoms were able to be manipulated to exhibit identical starting and ending points, as compared to the experimental data, as well as similar randomized speckle and bead scattering patterns. In addition, the amplitudes due to the surface, the glass bead, and a large set of microscopic, unresolvable scatterers remained consistently similar to the experimental data across all simulations (Fig. 4.1).

The perpetually small difference between the area below the experimental and simulated scattering due to the glass bead also implied that the simulations maintained significant a level of precision. When the simulated bead overlapped with the experimental bead, the average difference between the two traces was $2.7 \cdot 10^{-7}$ a.u.*m, less than half the size of the difference between offset beads, $6.0 \cdot 10^{-8}$ a.u.*m. The variance between the overlaid and offset average differences indicates a significant distinction between the two simulated

phantoms, demonstrating that, when overlaid, the simulated trace is similar in amplitude and pattern to the experimental trace. However, this method of comparison does suggest that the amplitude of the simulated bead may need to be slightly increased, to further mitigate the average difference.

The greatest point for concern when comparing the simulated and experimental data stems from the variable bead width demonstrated within the phantom. This increase in width by 0.28mm is a significant difference, especially when considering that the observed diameter of the bead from the experimental trace was 0.45mm. This indicates at least a 50% increase in simulated bead size (Fig. 4.4).

This particular result may have substantial consequences as this research moves towards training machine learning on simulated datasets. The goal in using machine learning, as will be further discussed in Section 5.1, is to train the algorithm to identify the size and position of a bead-like scatterer within a given RF trace. However, if this algorithm is taught to identify the size of a bead from simulations that consistently exaggerate the bead's width, it is possible that the program will face difficulties in accurately identifying the correct size when presented with experimental data.

The cause of the simulation's error, however, has not been made evident. One potential explanation may come from a recently observed effect when simulations were performed with significantly reduced speckle amplitudes. This response, which has been nicknamed "ringing," refers to a large dip in the trace, followed by a slow incline back to zero, observed after the bead's scattering (Fig 5.1). Much of the ringing is obfuscated in the presence of speckle, allowing the effect to remain largely unnoticed in simulations with realistic scaling of the generalized scatterers. In addition, since most of the comparisons were performed

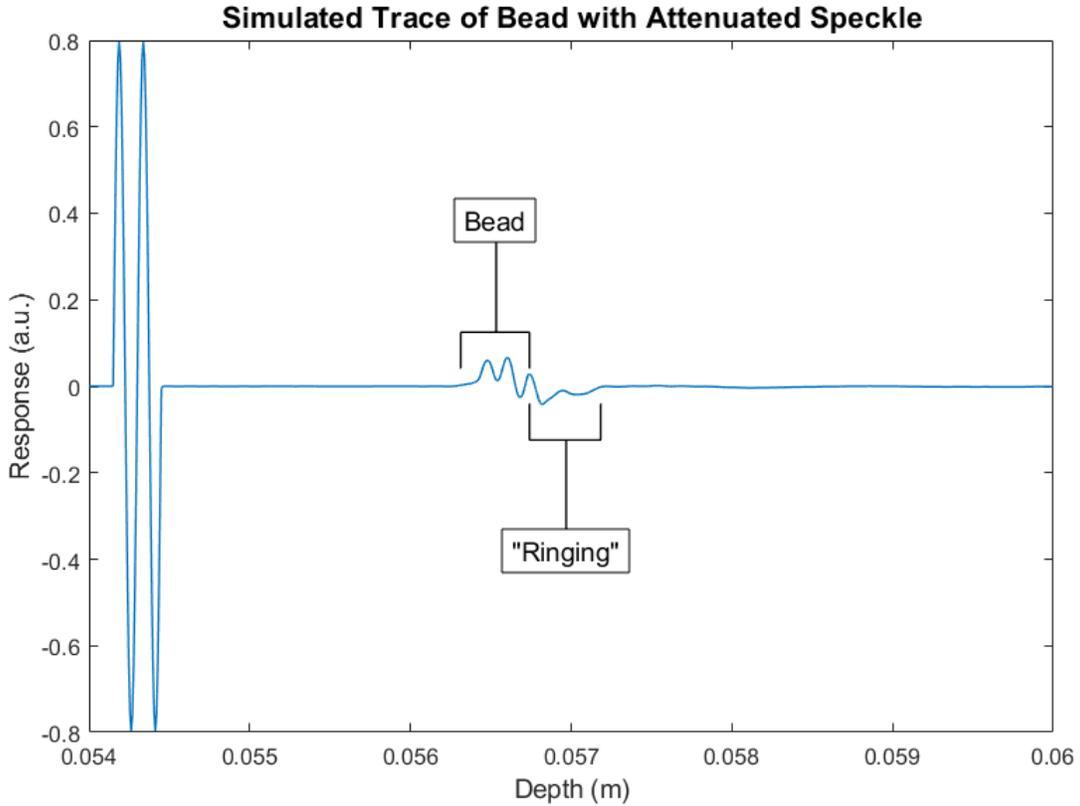


Figure 5.1: A simulated ultrasound trace with attenuated speckle to demonstrate the "ringing" effect. The estimated area of the trace containing the bead, as well as the area containing ringing, has been labeled. The width of the bead ($X=0.05631$ to $X=0.05674$) and the ringing ($X=0.05674$ to $X=0.05717$) are both approximately 0.43mm.

using the squared amplitudes, the dip would have appeared as a final positive peak due to the bead's scattering.

This theory holds up when estimating the true size of the bead by disregarding the section that follows a ringing pattern. By not including the ringing section, the bead widths were consistently estimated to be between 0.43-0.46mm, extremely close to the 0.45mm width input into the computation phantom. Similarly, the large negative peak due to the ringing fell between 0.15-0.25mm in width, though the smaller aftereffects of the ringing continued for around 0.2mm. This suggests that the main depression can account for the 50% increase

in bead width, with the small aftereffects likely being covered by the speckle in regular simulations. However, while the ringing effect neatly explains the consistently oversized bead width, further study is required to understand the source of the ringing.

Another possibility to examine in analyzing the discrepancies between the experimental and simulated traces is that the makeup of the experimental phantom may be playing a larger role than previously considered. The simulated bead's trace scatters as if it were hitting a slightly softer, heterogenous surface, as one would normally expect in a tissue sample. While microscopic glass beads dispersed throughout the phantom have been noted to mimic small structures within tissue reasonably well, large, hard scatters are often too homogenous to create an accurate tissue substitute [20].

In addition, attempting to create a phantom to mimic hard tissue may have been contrary to the initial objective of this research. While malignant tumors are noticeably more dense than normal, healthy tissues, they are still soft as compared to non-organic materials. For reference, the speed of sound within tumorous tissue has been characterized on the order of 1500m/s, whereas the speed of sound within glass is significantly larger, at 5000-6000m/s [20, 21]. If the purpose of training a machine learning algorithm is to use it to identify small abnormalities in tissue structure, using experimental phantoms based on glass beads may be counterproductive. Thus, in future studies, it would be prudent to perform experiments using other tissue substitutes, such as tapioca beads, rather than glass beads.

5.1 Future Work with Machine Learning

While the existence of the ringing effect continues to drastically increase bead size, we have decided to continue forward and initiate the machine learning process. It is expected that this process will take a significant amount of time, so while the foundations are being laid for machine learning, improvements will continue to be made on the simulations to ensure the best possible results.

There exists a large variety of algorithms for machine learning, each with strengths and weaknesses in regards to types of computations they can perform. In general, there are two main classes of machine learning: classification, which is used to sort data into categories, and thereby creating a discrete output, and regression, which is used to model continuous functions, and thus creates a continuous output. As such, in choosing a particular method for machine learning, the forms of both the desired the input and output data must inform the decision.

In the case of scatterer identification and placement, ultrasound data presents a difficult choice in algorithm selection. The desired output is currently a set of values describing the x-position (depth) and radius of the main scatterer. This would fall under the umbrella of a classification problem, since the desired output is discrete. However, classification works best with either discrete datasets as an input, or images. While it is possible to construct B-mode images from the simulated RF traces, each image would require between 50-100 traces, which would drastically increase the time and computational power needed to create the machine learning training set. By using RF data, however, the input dataset would be taking the form of a continuous function, which would be better suited as the input of a

regression model.

Essentially, the input data from our simulations is in the form required for regression machine learning, but the desired output should be in the format expected from a classification algorithm. Therefore, it is necessary to look beyond the most basic models in machine learning, and instead investigate more complex algorithms, such as neural networks, that can handle differentiated input and output data formats.

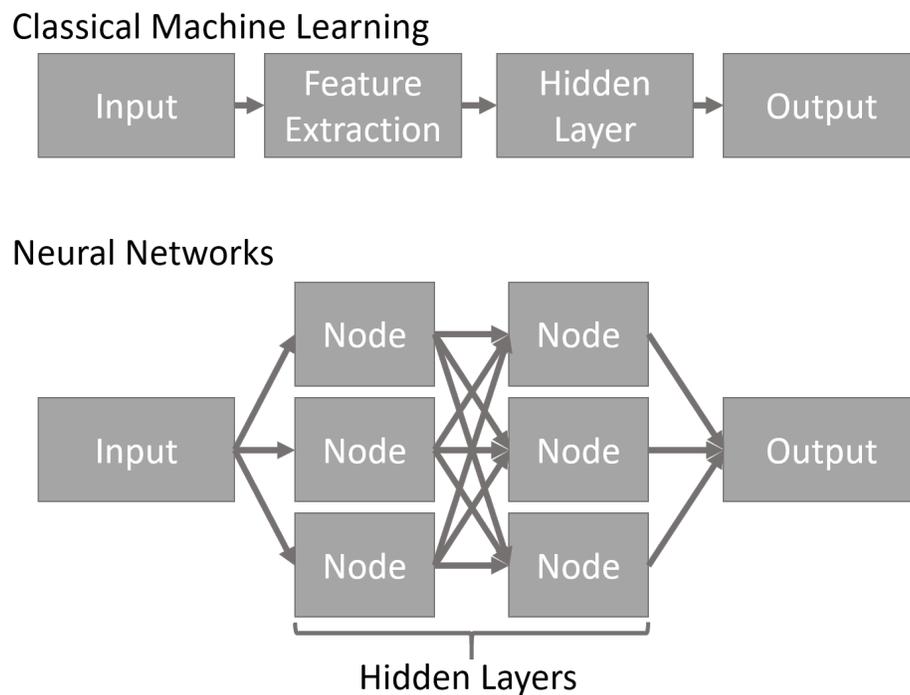


Figure 5.2: A diagram illustrating the procedural differences between classical machine learning and neural networks.

Neural networks vary from classical machine learning though the way in which they interact with and interpret data. Classical machine learning has four linear steps: input and feature extraction, which are constructed by the user, a hidden layer for computations, and an output. Neural networks, on the other hand, feature only one step for the user, the input, and then multiple hidden layers constructed from interconnected nodes perform both

feature extraction and computations, before sending the results to an output (Fig. 5.2). As such, the user is not required to delineate the desired features from the datasets, and instead allows the hidden layers to recognize patterns and determine the important parameters. In addition, neural networks are generally not restricted to specifically continuous or discrete datasets, demonstrating the flexibility required for this project.

Even within the narrowed field of neural networks, there are still a number of algorithms to choose from, Currently, we propose that the most appropriate method for our research will come in the form of convolutional neural networks (CNNs). CNNs are a powerful machine learning tool, generally used in image classification. However, unlike many classification algorithms, CNNs require very little pre-processing, and are able to take both the spacial and temporal characteristics of images into account. The latter trait will likely prove useful in analyzing ultrasound traces, which are intrinsically linked to time and space via the equations for motion (Eq. 3.1, 3.2).

Having determined a promising method for machine learning, the next few steps of this project can be outlined. First, the codes presented in Appendices B and C will be combined with simulation and phantom programs designed by another Herd Lab undergraduate researcher, Nguyen Nguyen. Her programs were designed in a similar fashion to those presented in this paper, but rather than focus on the addition of three dimensional object within the phantoms, her work implements the possibility of moving the simulated and transducer and performing scans of the same computational phantom from multiple positions. By combining both codes, the completed program will have the ability to both insert three dimensional beads and view the beads from different positions.

With the completed simulations, the formation of the machine learning training datasets

will begin. These datasets will be the result of repeatedly running the simulations (presumably on the order of 1000 iterations), with each simulation containing a single bead scatterer of random size between 0.5mm-3.5mm in diameter, placed at a random depth within the phantom. The training data will be split evenly between focused and unfocused transducers, and each dataset will include the simulated trace, along with the depth and size of the bead. The CNN will be trained with these sets as the input, and then validated by testing it on another collection of simulated datasets. If the algorithm performs well with the validation set, it will then be evaluated against sets of experimental traces. The training and testing of the algorithm, as a whole, will likely be long and iterative process, but it is heartening to have made another significant advancement in the scheme of this research.

References

- [1] D. Kane, “A brief history of musculoskeletal ultrasound: ‘From bats and ships to babies and hips’,” *Rheumatology*, vol. 43, no. 7, pp. 931–933, May 4, 2004, ISSN: 1460-2172. DOI: 10.1093/rheumatology/keh004.
- [2] T. Szabo, *Diagnostic Ultrasound Imaging: Inside Out*. Academic Press, Dec. 2013, ch. 1, 8, pp. 1–28, 213–242, ISBN: 978-0123964878.
- [3] A. Ng and J. Swanevelder, “Resolution in ultrasound imaging,” *Continuing Education in Anaesthesia Critical Care & Pain*, vol. 11, no. 5, pp. 186–192, Oct. 2011, ISSN: 17431816. DOI: 10.1093/bjaceaccp/mkr030.
- [4] H. Yusuf, “Modeling Speckles for Ultrasound Imaging,” Mount Holyoke College, 2015, p. 60.
- [5] C. Chinowsky, “Computational and experimental modeling of speckle for diagnostic ultrasonic imaging,” Mount Holyoke College, 2016, p. 76.
- [6] J. Gathany, Center for Disease Control, 1998. [Online]. Available: <https://phil.cdc.gov/Details.aspx?pid=1666>, (accessed: 03.28.2019).
- [7] K. J. Parker, “Superresolution imaging of scatterers in ultrasound B-scan imaging,” *The Journal of the Acoustical Society of America*, vol. 131, no. 6, pp. 4680–4689, Jun. 2012, ISSN: 0001-4966. DOI: 10.1121/1.4714341.
- [8] C. B. Burckhardt, “Speckle in ultrasound B-mode scans,” *IEEE Transactions on Sonics and Ultrasonics*, vol. 25, no. 1, pp. 1–6, Jan. 1978, ISSN: 0018-9537. DOI: 10.1109/T-SU.1978.30978.
- [9] J. G. Abbott and F. L. Thurstone, “Acoustic Speckle: Theory and Experimental Analysis,” *Ultrasonic Imaging*, vol. 1, no. 4, pp. 303–324, Oct. 1979.
- [10] E. J. Chen, W. K. Jenkins, and W. D. O’Brien, “Performance of ultrasonic speckle tracking in various tissues,” *The Journal of the Acoustical Society of America*, vol. 98, no. 3, pp. 1273–1278, Sep. 1995, ISSN: 0001-4966. DOI: 10.1121/1.414453.
- [11] F. Aalamifar, H. Rivaz, J. J. Cerrolaza, J. Jago, N. Safdar, E. M. Boctor, and M. G. Linguraru, “Classification of kidney and liver tissue using ultrasound backscatter data,” presented at the SPIE Medical Imaging, J. G. Bosch and N. Duric, Eds., Orlando, Florida, United States, Mar. 17, 2015, p. 94190X. DOI: 10.1117/12.2082300.
- [12] R. Wagner, S. Smith, J. Sandrik, and H. Lopez, “Statistics of Speckle in Ultrasound B-Scans,” *IEEE Transactions on Sonics and Ultrasonics*, vol. 30, no. 3, pp. 156–163, May 1983, ISSN: 0018-9537. DOI: 10.1109/T-SU.1983.31404.

- [13] D. R Foster, M. Arditi, F. S Foster, M. S Patterson, and J. W Hunt, “Computer simulations of speckle in b-scan images,” *Ultrasonic imaging*, vol. 5, pp. 308–30, Nov. 1983. DOI: 10.1177/016173468300500403.
- [14] J. S. Daba, J.-P. Dubois, and P. Jreije, “Advanced Stochastic Models for Partially Developed Speckle,” *International Journal of Electronics and Communication Engineering*, vol. 2, no. 5, p. 5, 2008.
- [15] J. A. Jensen, “FIELD: A program for simulating ultrasound systems,” *Medical and Biological Engineering and Computing*, vol. 34, pp. 351–352, Jan. 1, 1996.
- [16] MATLAB, *version 9.5.0.944444 (R2018b)*. Natick, Massachusetts: The MathWorks Inc., 2018.
- [17] J. A. Jensen, “Users’ guide for the Field II program,” p. 70, May 2014.
- [18] J. Jensen and N. Svendsen, “Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers,” *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, vol. 39, no. 2, pp. 262–267, Mar. 1992, ISSN: 0885-3010. DOI: 10.1109/58.139123.
- [19] J. A. Jensen and P. Munk, “Computer Phantoms for Simulating Ultrasound B-Mode and CFM Images,” in *Acoustical Imaging*, S. Lees and L. A. Ferrari, Eds., vol. 23, Boston, MA: Springer US, 1997, pp. 75–80, ISBN: 978-1-4613-4640-1 978-1-4419-8588-0. DOI: 10.1007/978-1-4419-8588-0_12.
- [20] M. O. Culjat, D. Goldenberg, P. Tewari, and R. S. Singh, “A Review of Tissue Substitutes for Ultrasound Imaging,” *Ultrasound in Medicine & Biology*, vol. 36, no. 6, pp. 861–873, Jun. 2010, ISSN: 03015629. DOI: 10.1016/j.ultrasmedbio.2010.02.012.
- [21] A. Keshavarzi, S. Vaezy, P. J. Kaczkowski, G. Keilman, R. Martin, E. Y. Chi, R. Garcia, and V. Y. Fujimoto, “Attenuation coefficient and sound speed in human myometrium and uterine fibroid tumors,” *Journal of Ultrasound in Medicine*, vol. 20, no. 5, pp. 473–480, May 2001, ISSN: 02784297. DOI: 10.7863/jum.2001.20.5.473.

Appendices

Appendix A

MATLAB Program for Processing Experimental Data

```
1 %Making_Bmode_images_nb.m
2 %Naomi Brandt
3 %Processes raw ultrasound data
4 %Last edited 4/2/2019
5
6
7 % Input parameters:
8 Cw = 1491; %'Enter sound speed of water (m/s):
9 Cs = 1500; %('Enter sound speed of sample (m/s): ');
10 FilePath = 'C:\Users\brand\Documents\College Files\2018-2019\
    HerdLab\'; % path to data files
11 FileNameBase = 'TEK0001';%filename base
12 winsize = 4; % window size in microseconds
13 numFrame = 7; % number of raster scan points
14 stepSize = 235; % number of steps between scans .00025 inch per
    step
15
16 % Load the data
17 rawData1=csvread('TEK00001.CSV');
18
19
20 k=char(50);
21 l=1;
22 FileName = strcat(FileNameBase,k, '.CSV');
23 firstdata = [];
24 fullFile=strcat(FilePath,FileName);
25 firstdata = csvread(fullFile);
26 amountdata = double(length(firstdata));
27
```

```

28 sampleData = zeros(amountdata,2,numFrame);% 'int32');
29 sampleData(:,:,1) = sampleData(:,:,1) + firstdata(:,:,);
30
31 VectorArray = zeros(amountdata,numFrame);% define vector array.
32 BmodeArray = zeros(amountdata,numFrame);% define Bmode Array.
33
34 %Loads multiple datasets at once
35 for i = 1:numFrame
36     j = char(49+i);
37     FileName = strcat(FileNameBase,j, '.CSV');
38     fullFile=strcat(FilePath,FileName);
39     sampleData(:,:,i) = sampleData(:,:,i)+ csvread(fullFile);
40 end
41
42 VectorArray = squeeze(sampleData(:,2,:));
43
44 BmodeArray = URllogenv(VectorArray);
45 %Bmode vector away is log scale conversion of vector array
46
47 XStep = stepSize*.0025*2.54; % step between scan in mm
48 %XDistance = zeros(numFrame);
49 for i = 1:numFrame
50     XDistance(i) = XStep*i;
51 end
52
53 %Calculate depth into sample
54 YDepth = (winsize*.000001)* Cs*100/2*1/100;
55 YDepth=rawData1(:,1)*Cw/2;
56
57
58 % Plot amplitude vs depth
59 figure
60 plot(YDepth, rawData1(:,2))
61 title('Experimental Trace of ~1mm Bead')
62 ylabel('Response (a.u.)')
63 xlabel('Depth (m)')
64 axis([.054,.06,-1,.8])

```

Appendix B

MATLAB Program for Designing Computational Phantom

```
1 %bead_phantom_nb.m
2 %Naomi Brandt
3 %Creates ultrasound phantom
4 %Last Edited 4/5/2019
5
6 function [ positions , amp ] = bead_phantom_nb( N )
7
8 %Creating phantom base
9 x_size = 35/1000; % Width of phantom [m]
10 y_size = 35/1000; % Transverse width of phantom [m] - produces
    transverse movement
11 z_size = 10/1000; % Height of phantom [m]
12 z_start = 54/1000; % Start of phantom surface [m];
13
14
15 %Create the random general scatterers
16 x = (rand (N,1) -0.5)*x_size;
17 y = (rand (N,1) -0.5)*y_size;
18 z = rand (N,1)*z_size + z_start;
19
20 % Generate the amplitudes with a Gaussian distribution
21 ampStart =randn(N,1);
22
23
24 % Make scatterer and set the amplitudes
25
26 r=(.45/2)/1000; % Radius of cyst [m]
27 xc=0/1000; % Place of cyst [m]
28 zc=4.01/1000+z_start;
```

```

29
30 %Define inside the bead and amplify amplitudes
31 inside = ( ((x-xc).^2 + (z-zc).^2 ) < r^2);
32 amp = ampStart.*.4.*10^22.*(1-inside)+ 2.945.*10^22.* inside;
33
34 % Place the initial point scatterers (surface) in the phantom
35 x(1) = 0/1000;
36 y(1) = 0;
37 z(1) = z_start;
38 amp(1) = 0;
39
40 x(2) = 0/1000;
41 y(2) = 0;
42 z(2) = z_start+1.6/1000;
43 amp(2) = 75*10^22;
44
45 % Place 0 amp scatterers to make simulation have no scattering
    before surface and make all scatters before then actually
    start at the surface
46 for n=1:N
47     if z(n)<z_start+1.6/1000
48         z(n)=z_start+1.6/1000;
49         amp(n)=0;
50     end
51 end
52
53
54 % Return the variables
55 positions=[x y z];
56
57 pos_amp_inside=[x y z amp inside];
58
59 end

```

Appendix C

MATLAB Program for Performing Simulations

```
1 %bead_sim_nb.m
2 %Naomi Brandt
3 %Runs field-ii simulation
4 %Last edited 4/2/2019
5
6 field_init
7
8 % Set initial parameters
9 f0=5e6; % Transducer center frequency [Hz]
10 fs=100e6; % Sampling frequency [Hz]
11 c=1491.23; % Speed of sound [m/s]
12 cs=1500; %speed of sound in sample (m/s)
13
14 lambda=c/f0; % Wavelength [m]
15 R=3/1000; % Radius of transducer in m
16 Rfocal=52.2/1000; % Focal radius of transducer in m
17 ele_size=3/1000; % Size of mathematical elements in m
18
19 kerf= 1/1000;
20 focus = [.05*50/1000,.05*50/1000,.05*50/1000 + 15/1000];
21
22 % Define the transducer
23 Th = xdc_concave(R, Rfocal, ele_size);
24
25 %Import phantom scatters
26 %Increase N to increase random general scatterers
27 [phantom_positions, phantom_amplitudes] = bead_phantom_nb
    (100000);
28
```

```

29 % Set the impulse response and excitation of the emit aperture
30 impulse_response=sin(2*pi*f0*(0:1/fs:2/f0));
31 impulse_response=shiftdim(impulse_response,1);
32 windowHanning = hanning(max(size(impulse_response)));
33 %size(windowHanning)
34 %size(impulse_response)
35
36 impulse_response_windowed=impulse_response.*windowHanning;
37 excitation=sin(2*pi*f0*(0:1/fs:2/f0));
38 xdc_excitation (Th, excitation);
39
40 % Do the calculation
41 %Procedure for calculating the received signal from a
    collection of scatterers and for each of the elements in the
    receiving aperture
42 [v,t]=calc_scatt_multi (Th, Th, phantom_positions,
    phantom_amplitudes);
43
44 % Process the individual response, change time array to depth
    array
45 N=length(v());
46 Ts = 1/fs;
47 Ds=(Ts*cs/2); %define distance in m
48 d_start = (t*cs/2); %define start distance in m
49 d_array = zeros(N,1);
50 for i = 1:N
51 d_array(i) = d_start + Ds*(i-1);
52 end
53
54 sim_size=size(v);
55
56 %Plot the trace as amplitude versus depth
57 figure
58 plot(d_array,v)
59 title("Simulated Trace")
60 ylabel('Response (a.u.)')
61 xlabel('Depth (m)')
62 axis([.054,.06,-1,.8])

```

Appendix D

MATLAB Program for Comparing Datasets

```
1 % sim_exp_comparison.m
2 %Naomi Brandt
3 %Comparing simulated and experimental data
4 %Last edited 4/5/2019
5
6
7 %Making simPoint and expPoint, which are just focused around
   the simulated and experimental bead
8 simPoint=[];
9 expPoint=[];
10 diffAvg=[];
11 iter=10; %number of iterations to run through
12
13 %Run experimental trace processing
14 run Making_Bmode_images_nb.m
15
16 for i=1:iter
17     run bead_sim_nb.m
18
19     %squaring simulated and experimental signals
20     sim_vsqr=v.^2;
21     exp_vsqr=rawData1(:,2).^2;
22
23     %making matrices of simulated and experimental data
24     simDat=[sim_vsqr,-d_array];
25     expDat=[exp_vsqr,-YDepth];
26     eS=size(expDat(:,1));
27     sS=size(simDat(:,1));
28
```

```

29     %Determining min and max of the simulation and experimental
        depths
30     simMinMax=[min(simDat(:,2)),max(simDat(:,2))];
31     expMinMax=[min(expDat(:,2)),max(expDat(:,2))];
32
33     %Deleting all simulation points that are out of
        experimental range
34     simCut=[];
35     for i =1:sS(1,1)
36         sim = simDat(i,2);
37         if sim >= expMinMax(1,1) && sim<=expMinMax(1,2)
38             row = find(simDat(:,2) == sim);
39             simCut=[simCut;simDat(row,:)];
40         end
41     end
42
43     cutS=size(simCut(:,1));
44     expRange=expMinMax(1,2)-expMinMax(1,1);
45
46     %Binning exp and sim data and adding bins to data matrices
47     %Note: increase bin size for more of a point by point
        comparison, decrease for more of a general shape
        comparison
48     [simN, simEdges, simBins]=histcounts(simCut(:,2),500);
49     simCut=[simCut,simBins];
50     [expN, expEdges, expBins]=histcounts(expDat(:,2),500);
51     expDat=[expDat,expBins];
52
53     %Averaging values within each bin
54     simAvg = accumarray(simCut(:,3),simCut(:,1))./accumarray(
        simCut(:,3),1);
55     expAvg = accumarray(expDat(:,3),expDat(:,1))./accumarray(
        expDat(:,3),1);
56
57     comp=(expAvg+simAvg)/2;
58     depthMid =((expEdges(1:end-1)+expEdges(2:end))/2).';
59
60     expAvg=[expAvg,depthMid];
61     simAvg=[simAvg,depthMid];
62
63     % Centering sim and experimental data on the bead
64     for i=1:size(simAvg(:,1))
65         cut=simAvg(i,2);
66         if cut<=-.0555 && cut>= -.0585
67             simPoint=[simPoint;simAvg(i,:)];

```

```

68     end
69 end
70
71 for i=1:size(expAvg(:,1))
72     dat=expAvg(i,2);
73     if dat<=-.0555 && dat>= -.0585
74         expPoint=[expPoint;expAvg(i,:)];
75     end
76 end
77
78 %Finding the difference between the two averaged traces
79 diffAvg=[diffAvg,(expAvg(:,1)-simAvg(:,1))];
80
81 end
82
83 %Find the mean difference
84 av=[]
85 for i=1:iter
86     av=[av,mean(diffAvg(:,i))]
87 end
88
89 %lines 103-112: plot all iterations
90 figure
91 xlabel('Depth (m)')
92 ylabel('Response (a.u.)')
93 axis([.0555,.0585,0,.01])
94 hold on
95
96 for i=1:201:(i*201)
97     plot(-simPoint(i:i+200,2),simPoint(i:i+200,1),'c')
98 end
99
100 plot(-expPoint(1:201,2),expPoint(1:201,1),'r')
101
102 %Find overall average difference around bead
103 TdiffAvg=sum(abs(diffAvg(202:403,1)))/(403-202)
104
105 %Plot single comparison and the average
106 figure
107 plot(-expAvg(:,2),expAvg(:,1),'r',-simAvg(:,2),simAvg(:,1),'b'
    , -depthMid(:,1),comp(:,1),'m')
108 axis([.056,.0575,0,.009])

```

Appendix E

Table of Overlaid and Offset Simulated Areas and Differences

Overlaid Area	Overlaid Area Difference	Offset Area	Offset Area Difference
6.85E-07	9.19E-09	2.10E-07	4.84E-07
6.12E-07	8.16E-08	1.47E-07	5.47E-07
5.53E-07	1.41E-07	1.29E-07	5.65E-07
5.42E-07	1.52E-07	1.26E-07	5.68E-07
5.23E-07	1.71E-07	1.21E-07	5.73E-07
5.13E-07	1.81E-07	1.20E-07	5.74E-07
5.06E-07	1.88E-07	1.19E-07	5.75E-07
4.61E-07	2.33E-07	1.13E-07	5.81E-07
4.34E-07	2.60E-07	1.03E-07	5.90E-07
4.32E-07	2.62E-07	9.61E-08	5.98E-07
4.01E-07	2.93E-07	8.85E-08	6.05E-07
3.72E-07	3.22E-07	7.59E-08	6.18E-07
3.64E-07	3.29E-07	6.43E-08	6.30E-07
3.39E-07	3.55E-07	6.13E-08	6.33E-07
3.06E-07	3.88E-07	5.81E-08	6.36E-07
2.99E-07	3.95E-07	5.79E-08	6.36E-07
2.98E-07	3.95E-07	4.53E-08	6.49E-07
2.91E-07	4.03E-07	4.18E-08	6.52E-07
2.78E-07	4.16E-07	1.86E-08	6.75E-07
2.48E-07	4.46E-07	1.73E-08	6.77E-07