Displacement Data Processing for ARFI Imaging

by

Jude Onyia
Bachelor of Engineering (Honours),
University of Wollongong, Dubai, UAE,
2019

A Report Submitted in Partial Fulfillment of the Requirements for the Degree of

MASTER OF ENGINEERING

in the Department of Electrical and Computer Engineering

SUPERVISORY COMMITTEE

Displacement Data Processing for ARFI Imaging

by

Jude Onyia
Bachelor of Engineering (Honours),
University of Wollongong, Dubai, UAE,
2019

Supervisory Committee

Dr. Daler Rakhmatov, **Supervisor**
Department of Electrical and Computer Engineering

Dr. Mihai Sima, **Departmental Member**
Department of Electrical and Computer Engineering

ABSTRACT

Shear Wave Elastography (SWE) is an ultrasound imaging application that aims to estimate bio-mechanical properties of tissues, such as the Young's modulus, within an imaged area of interest. The SWE process entails tracking tissue movement caused by an internal or external excitation, with the goal of determining the shear wave speed (SWS) propagating through the tissue. Acoustic radiation force impulse (ARFI) imaging is an important SWE modality that involves creating an internal push via a focused ultrasound transmission, which gives rise to shear waves, followed by tissue displacement tracking via ultrafast plane-wave transmissions.

This report deals with displacement data processing for ARFI imaging that aims to provide SWS estimates. Our reported evaluation results are based on the public-domain ARFI dataset from the UltraSound ToolBox (USTB), serving as an illustrative example. This dataset contains ultrasound scans of a tissue-mimicking phantom containing an 80-KPa sphere (10 mm in diameter) embedded in the 25-KPa background.

This report discusses three well-known displacement tracking techniques: Doppler frequency estimation based on auto-correlation, two-dimensional analytic minimization (2D AM), and clutter filter wave imaging (CFWI). The SWS values were then computed using the sliding-window Radon sum transform, applied to the displacement data after directional filtering and masking based on Canny edge detection. The latter highlights relevant spatio-temporal trajectories in displacement images that increases the accuracy of SWS estimates. Additionally, the SWS images themselves underwent morphological opening that enhanced their visual appearance. Overall, this report recommends using the auto-correlation technique that led to SWS estimates in good agreement with their ground-truth values.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to acknowledge the University of Victoria and its Electrical and Computer Engineering department for living up to its reputation and providing a safe and wonderful learning experience. I would like to thank Dr. Daler Rakhmatov for his diligent guidance through this project, and Dr. Mihai Sima for being a supervisory committee member. I would like to credit the Ultrasound ToolBox developers and others for providing the dataset used, and I would like to thank authors such as Hassan Rivaz, Ole Marius Hoel Rindal, Sebastien Salles, Ned Rouze, and others whose works inspired the efforts in this project.

DEDICATION

I dedicate this report to my family members for supporting my studies overseas, and my professors and peers for a great university experience.

1. INTRODUCTION

The relative state of tissue in terms of its pathology and function is inextricably linked to its bio-mechanical properties, such as the shear or elastic moduli (Young's modulus), Poisson's ratio, and the tissue's normal or shear strain response to applied load [1]–[3]. The process of extracting these bio-mechanical properties to detect pathological tissue alterations is referred to as *elastography* [4]. Shear wave elastography involves extracting these bio-mechanical properties by tracking tissue deformation caused by internal or external stimuli. Innovative approaches to shear wave elastography using ultrasound or magnetic resonance (MR) have emerged in the past two decades [1], [3], [5]. The use of heat, water jets, vibration shear waves, quasi-static compression, and dynamic compression as stimuli to insight motion in tissue has been thoroughly investigated [1]–[5].

This report focuses on studying shear wave elastography approaches that utilize acoustic radiation force impulse (ARFI) excitations as stimuli, and uses ultrasound plane-wave pulse-echo acquisitions for tracking tissue deformation. This chapter of the report gives a thorough description of the process involved in ARFI elastography by first defining ultrasound imaging, and the improvements in its data acquisition procedure (ultrafast plane wave imaging), which makes shear wave elastography possible. Different works on shear wave elastography are briefly mentioned, leading up to a discussion of the contributions of this work.

1.1. Ultrasound Imaging

Ultrasound imaging has been held in high regard in areas of biomedical diagnostics due to its excellent temporal resolution, good image quality, and minimal invasive nature [6]–[8]. It has been considered the *golden-standard* for diagnosing diseases in the heart and liver [6]. The cost effectiveness and real-time nature of ultrasound imaging has been proven highly valuable

in emergency settings, rural clinics, and developing countries [7], [8]. Typical pulse-echo ultrasound imaging is a process that involves exposing an area of interest (AOI) to ultrasonic pulses, recording the backscattered echoes, and using that data to produce an image of the internal structure of the AOI [6]–[10]. While different applications require different produced images, ultrasound is most commonly associated with getting a Brightness mode (B-mode) image, where individual pixels represent the intensity of received echoes. Motion mode (M-mode) images are required if one intends to capture the dependency on time of a particular section of the recorded data [7].

At first, the ultrasonic echoes were collected from multiple focal points synthesized dynamically along a single beam line (or scan line) [8], [10]. Multiple scan lines were acquired by mechanically moving the focused transducer. The development of multiple element arrays and electronic focusing allowed for line-per-line acquisition without the mechanical movement, however, this process still proved to be too slow for applications requiring high frame rates (such as in shear wave elastography) or requiring large field-of-view images (in the order of centimeters) [9], [10].

Compressed sensing algorithms were developed to allow a reduction of transmit events, hence, achieving higher frame rates [6]. Compressed sensing is a data acquisition technique that assumes certain classes of signals, such as natural images, have a sparse representation, and hence can be reconstructed from a small number of linear measurements [11]. The signals are recovered by solving a nonlinear optimization problem [11]. Although compressed sensing does well in attempting to address the down-sampling artifacts caused by reducing the number of transmit events, ultrasound specific properties deteriorate its performance [6]. This is because ultrasound images are rarely sparse due to their characteristic speckles [6].

The trade-off between frame rate, resolution and imaging depth should be considered and weighed for importance depending on the application [7]. Resolution can be improved by using higher pulse modulation frequencies in the transmission of ultrasonic pulses. However,

this will cause shorter wavelengths that will be absorbed closer to the surface, hence, reducing imaging depth [7]. Currently, higher frame rates are achieved by using alternatives to line-per-line acquisition that drastically reduce the number of transmit events while retaining an acceptable resolution.

Very high frame rates (in the order of hundreds or even thousands of frames per second) can be achieved through unfocused transmission schemes based on plane or diverging waves [7]. Compared to line-per-line focused beam transmissions, using plane or diverging waves results in loss of lateral resolution [7]. Justifying this loss in resolution against gaining higher frame rates is application dependent. The ultrafast image acquisition rates made possible by plane wave ultrasound imaging enabled new biomedical applications to emerge [8], [10]. Applications such as ultrasound-based blood flow characterization performed in [12], and tissue motion measurement performed in [13]. It also improved the understanding of more complex dynamics found in echocardiography (heart motion analysis), ultrafast Doppler imaging, ultrafast vector flow imaging, and some 3-D imaging applications (such as shear wave elastography) [11].

Unlike focused beam transmission schemes that involve transmitting focused pulses and collecting data one scan line at a time, plane wave ultrasound imaging involves insonifying the entire AOI with one or a few transmitted waves, as shown in Figure 1.1. The backscattered echoes are beamformed into synthetic scan lines that can be processed in parallel, allowing for fast signal acquisition and processing [8], [11]. Through plane wave ultrasound imaging, the frame rate is limited only by the propagation of the ultrasound signals through the medium [11]. This allows ultrasound imaging systems to achieve frame rates in the order of kilohertz or higher [10], [11]. One can collect backscatter echo data for a large-AOI image from a single plane wave emission in the same acquisition time as that of a single scan line for a focused beam transmission scheme at the expense of deteriorated resolution [7]. Coherent compounding is a technique used to improve resolution and reduce noise, where multiple tilted plane waves transmitted at different angles result in multiple datasets that are coherently

combined to a final compounded ultrasonic image with improved quality [7], [8].

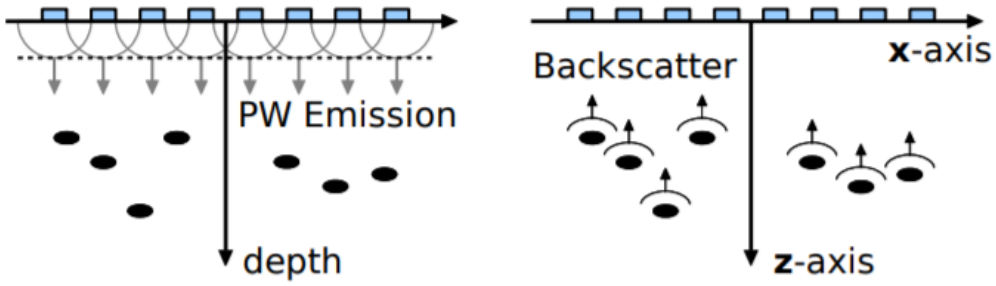


Figure 1.1: Plane Wave (PW) Ultrasound Imaging [8]

1.2. Shear Wave Elastography

Shear wave elastography typically employs ultrafast plane wave imaging to track tissue deformation and extract biomechanical properties to detect pathological tissue alterations [1]–[3], [5]. This begins with compressing the tissue using either quasi-static approaches (pressing the ultrasound probe against the tissue) or dynamic approaches (utilizing a low-frequency excitation) [5]. In quasi-static approaches, compression can either be applied as a single pressing motion (this is referred to as single-step shear) or as a series of press-and-release tissue deformations (this is referred to as cyclical shear) [1]. In dynamic approaches, tissue is compressed with either a transient tone burst of shear wave excitation produced from an acoustic radiation force (ARF) pulse, or harmonic pulses caused by an external device or an internal organ such as the heart [1]. The elastic tissue response can be captured using ultrafast plane wave imaging. It is assumed that the data acquisition time is significantly shorter than the time at which the tissue changes its state in-between plane wave emission and data acquisition cycles [3].

The need for the development of shear wave elastography stems from visibility problems in conventional ultrasound imaging [3], [10], [14], [15]. For example, tumors of the prostate or the breast may be poorly visible in standard (B-mode) ultrasound images, but due to the

viscoelastic behavior of the tumor differing from the surrounding tissue, shear wave elastography can be used to provide a visual depiction of the location of the pathological tissue [3], [10], [15].

As mentioned earlier, ultrafast imaging allows for data acquisition of the entire AOI from a single plane wave emission. Radio Frequency (RF) data samples are collected as a 2D matrix (or frame), where the vertical dimension corresponds to the ultrasound pulse round-trip time (also referred to as fast time), and the horizontal dimension corresponds to the lateral coordinates of sensing elements [6]. The plane wave emission and data acquisition cycle is repeated to create multiple frames that capture the motion of the tissue, resulting in a 3D data volume as shown in Figure 1.2 [1]–[5], [15]. This data is fed to a beamforming algorithm where synthetic scan lines are reconstructed simultaneously, producing beamformed in-phase/quadrature (IQ) data [6], [8].

Figure 1.2 shows the IQ data can be filtered and passed through a displacement (or motion) estimation algorithm that quantifies changes in the lateral and axial (depth) directions (x-axis and z-axis, respectively). There are two common strategies that displacement estimation algorithms follow. Window-based methods finds the displacements of each sample between frames by considering a window around the sample and assuming that displacement within the window is constant. An example of this approach is the popular Normalized Cross-Correlation (NCC) technique [5], [16]. Optimization-based methods utilize a regularized cost function to find the displacements between frames, which is more robust to signal decorrelation and out of plane motion [16]. Examples of these methods are the Dynamic Programming (DP) approach presented in [17], the GLUE method [18], the CGLUE method [4], and the 2D Analytic Minimization (AM) method [19]. These methods are further discussed in Chapter 2 of the report.

As shown in Figure 1.2, displacement estimation algorithms usually obtain displacement data from IQ data, however, some techniques such as the CGLUE method in [4] highlights

a benefit in obtaining displacement data from raw RF data. Displacement data is used to calculate the velocity of the shear wave in different sections or windows of the imaging field, producing a two-dimensional depiction of the shear wave speed (SWS) estimates of the region. These estimates can be further processed to get the Young's modulus by assuming soft tissue to be linear viscoelastic and isotropic with constant density, thus producing an elastogram representing the stiffness of different sections in the AOI [3], [15].

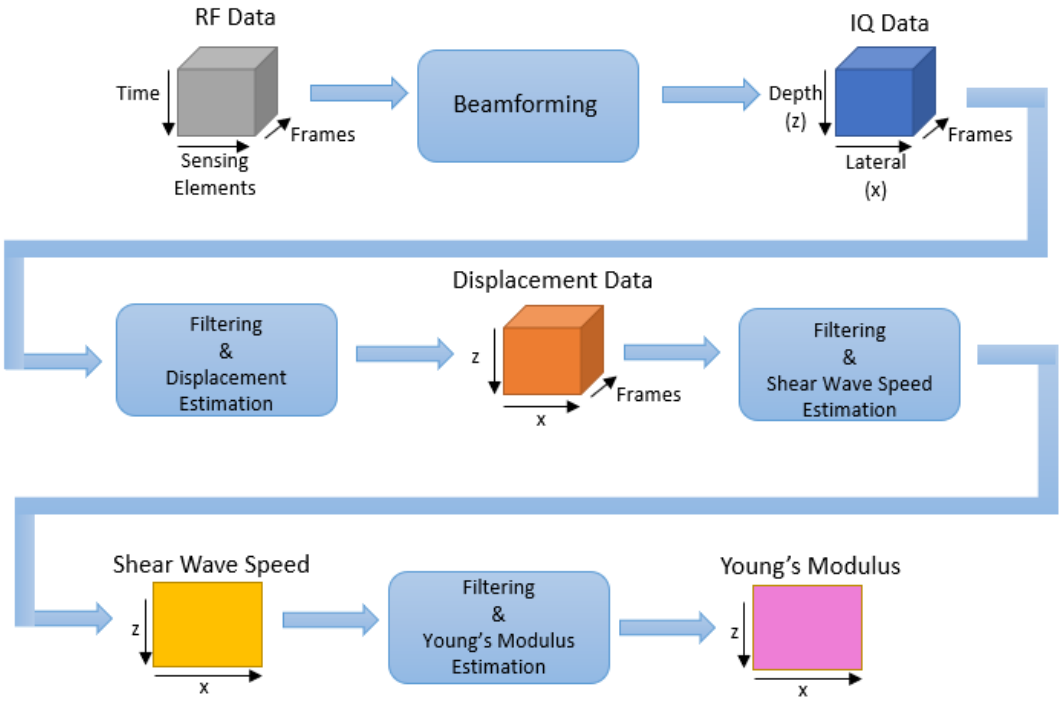


Figure 1.2: Elastography Data Processing pipeline

1.3. Report Contribution and Organization

This report conducts a visual comparison of different combinations of techniques involved in selected stages of the data processing pipeline shown in Figure 1.2. The influence of the beamforming method is outside the scope of this project. Our focus is on post-processing of the IQ data related to the *Filtering & Displacement Estimation* and the *Filtering & Shear Wave Speed Estimation* stages in Figure 1.2. For the purpose of this report, the performance evaluation of various investigated techniques end at producing a shear wave speed (SWS)

image. In other words, our main goal is to obtain SWS images indicating the presence of SWS variations, as opposed to trying to generate a final elastogram (Young's modulus) image based on SWS estimates.

The experimental cases investigated in the report will consider the beamformed IQ data as the input, obtained using the delay-and-sum (DAS) technique [20]. Chapter 2 discusses three candidate displacement estimation techniques, namely, the Clutter Filter Wave Imaging (CFWI), the Two Dimensional Analytic Minimization (2D AM), and a Doppler frequency estimation technique based on auto-correlation. Chapter 2 also discusses SWS estimation techniques such as the Linear Radon Transform (LRT) and the Radon Sum Transform (RST). Chapter 3 examines post-processing techniques of displacement estimates, such as the use of directional filtering and a Canny edge detection mask (coupled with the sliding-window RST). The latter is one of the contributions of this project and a unique strategy compared to other works. Another contribution of this work is the additional post-processing of the SWS image using morphological filtering, which is also discussed in Chapter 3. Chapter 4 compares results of different combinations of displacement and SWS estimation techniques under consideration. Chapter 5 provides a summary of the report and outlines possible directions for future work.

2. BACKGROUND

This chapter introduces concepts and works related to displacement estimation and shear wave speed (SWS) estimation. Detailed descriptions are provided, and the attributes and benefits of using these techniques are discussed.

2.1. Displacement Estimation

Three displacement estimation techniques are discussed in this section. The first method relies on a window-based estimation strategy that analyzes the correlation between neighboring data. The second method uses an optimization-based approach with a regularized cost function. The third method is a non-motion estimation approach that justifies the complexity of the other two methods and serve as an alternative.

2.1.1. UltraSound ToolBox (USTB)

The UltraSound ToolBox (USTB) is an open-source MATLAB (The Mathworks, Inc., Natick, MA, USA) toolbox for processing ultrasonic signals [21]. Developed by a diverse group of software developers and experts in the field of ultrasound, this toolbox features image reconstruction techniques, visualization tools and processing techniques to analyze and process ultrasonic records [21]. It aims to provide a common environment to compare algorithms, and encourages the publication of open-access datasets and research findings [21].

Displacement estimation functions implemented as part of the USTB are based on techniques derived for blood velocity measurement [20], [22]–[24]. These techniques are based on the direct relationship between the observed Doppler frequency spectra of echoes and velocities present within a tissue [22]–[24]. According to the Doppler effect, velocities are

encoded in the observed frequency shifts of back scattered echoes [24]. These frequency shifts with respect to the center frequency of the transmitted ultrasound pulses are referred to as Doppler frequency shifts [23], [24]. Therefore estimating the Doppler frequency shift of a small region leads to an estimate of the mean velocity within that region. Ideally, the region would be small enough that a single velocity would be present. However physical limitations prevent the analysis of a sufficiently small region [24]. Therefore, several works have presented methods of Doppler frequency estimation in the presence of multiple velocities. These works derived Doppler frequency estimators based on Doppler signal statistics [22]–[35].

Fundamentals on Velocity Estimation Based on Doppler Frequencies

The relationship between mean velocity and the observed Doppler frequency shift can be expressed as shown in Eq. 2.1, where c is the speed of sound in the area of interest (AOI), f_0 is the center frequency of transmitted ultrasound pulses, v is the mean velocity to be estimated, θ is the angle between the direction corresponding to the mean velocity and the direction of the transmitted pulses, and Δf is the mean Doppler frequency shift. Eq. 2.1 can be rewritten as Eq. 2.2, where the mean velocity is expressed as a function of the mean Doppler frequency shift.

$$\Delta f = 2 f_0 \left(\frac{v}{c}\right) cos\theta \quad [23] \tag{2.1}$$

$$v = \frac{c}{2cos\theta}\left(\frac{\Delta f}{f_0}\right) \quad [22], [23] \tag{2.2}$$

Frequency domain techniques are commonly employed to estimate the mean Doppler frequency shift by computing the power spectrum of the recorded signal [22]–[24], [27]–[35]. The observed Doppler frequencies are considered to be a Gaussian process, therefore their corresponding probability density function can be estimated by their power spectrum

[23]. The Fourier transform $\hat{s}(\omega)$ of the recorded signal is expressed in Eq. 2.3 and the power spectrum $S(\omega)$ is obtained in Eq. 2.4, where $*$ is the complex conjugate, and ω is the instantaneous Doppler angular frequency [30], [32], [34]. After normalization as shown in Eq. 2.5, the mean Doppler frequency shift Δf is obtained according to Eq. 2.6 and Eq. 2.7 [23], [24].

$$\hat{s}(\omega) = \int_{-\infty}^{\infty} s(t) e^{-i\omega t} dt \quad [23] \tag{2.3}$$

$$S(\omega) = \hat{s}^*(\omega)\hat{s}(\omega) \quad [23] \tag{2.4}$$

$$p(\omega) = \frac{S(\omega)}{\int_{-\infty}^{\infty} S(\omega) d\omega} \quad [22], [23] \tag{2.5}$$

$$\bar{\omega} = \int_{-\infty}^{\infty} \omega p(\omega) d\omega \quad [22], [23] \tag{2.6}$$

$$\Delta f = f_0 - \left(\frac{\bar{\omega}}{2\pi}\right) \quad [22] \tag{2.7}$$

Doppler Frequency Estimator Based on Auto-correlation

The Doppler frequency estimator introduced in [22] was chosen as one of the displacement estimation techniques compared in this report due to its relative simplicity and accuracy [20]. The mean Doppler angular frequency $\bar{\omega}$ from Eq. 2.6 can be obtained using an auto-correlation function $R(\tau)$ defined by Eq. 2.8 and Eq. 2.9. It allows $\bar{\omega}$ to be approximated with Eq. 2.10, where T is the inter-frame delay and ϕ is the phase of the auto-correlation function

given by Eq. 2.11.

$$R(\tau) \quad = \quad |R(\tau)|e^{j\phi(\tau)} \quad = \quad R_r(\tau) + jR_i(\tau) \quad [22] \tag{2.8}$$

$$R(\tau) = \int_{-\infty}^{\infty} S(\omega)e^{j\omega\tau}d\omega \quad [22] \tag{2.9}$$

$$\bar{\omega} = \frac{\phi(T)}{T} \quad [22] \tag{2.10}$$

$$\phi(T) = tan^{-1}\frac{R_i(T)}{R_r(T)} \quad [22] \tag{2.11}$$

Ultrasound data acquisition can be performed with either the continuous wave Doppler system or the pulsed Doppler system [22], [24]–[26], [31], [33], [35]. The continuous wave Doppler system is an ultrasonic data acquisition procedure that involves the transmitter continuously sending ultrasound waves and the receiver continuously recording the back-scatter [25]. The pulsed Doppler system is the procedure used in modern ultrasound applications where ultrasound pulses are emitted periodically [26]. The continuous wave Doppler system provides more information per depth on a given beam line, whereas the pulsed Doppler system can reach greater depths on a given beam line [22], [25], [26]. With a pulsed Doppler system acquisition, a low-pass filter can be used to obtain continuous time values needed for Doppler signal statistics. The auto-correlation $R(T)$ between samples from a pulsed Doppler acquisition can be defined in Eq. 2.12, where n is the number of transmit events in a given frame, and t is a variable for the travel time of the ultrasonic pulses.

$$R(T) = \int_{t-nT}^{t} s(t') \times s^*(t' - T)dt' \quad [22] \tag{2.12}$$

In the context of ARFI, however, the auto-correlation function $R(T)$ tracks displacements rather than frequency shifts, where T would be the time between consecutive displacement frames. Eq. 2.12 is approximated with Eq. 2.13 for discrete time computation, where m is the frame number [36]. For smaller displacements usually found in ARFI datasets, the auto-correlation methods typically perform well compared to normalized cross-correlation techniques and is less computationally expensive [22], [36].

$$\hat{R}(T) = \sum_{m-nT}^{m} s(m) \times s^{*}(m - T) \quad [36] \tag{2.13}$$

2.1.2. Two-Dimensional Analytic Minimization (2D AM)

The Analytic Minimization (AM) algorithm proposed in [19] is an optimization-based displacement estimation strategy. It minimizes a cost function that models similarity in data intensity and displacement discontinuities, allowing robustness towards small decorrelations in speckle tracking. The algorithm is also robust towards large local decorrelations due to its ability to exploit robust statistics by utilizing iterated re-weighted least squares (IRLS) to handle uncorrelated data as outliers [19].

Dynamic Programming (DP)

Denoting I_1 as a single beamformed data frame and I_2 as the following consecutive frame representing a deformed version of I_1, as shown in Figure 2.1, the goal of AM is to acquire a collection of $a_{i,j}$ and $l_{i,j}$ values representing axial and lateral displacements, respectively (where (i, j) represent lateral and axial positions of a tracked point, respectively) [19]. Dynamic Programming (DP), a technique presented by some of the same authors of [19] in a previous work described in [17], is first used to calculate integer values for displacements $a_{i,j}$ and $l_{i,j}$ by considering echo amplitudes and displacement continuity in beamformed data. For each scan line j (a column in Figure 2.1), a cost function is generated according to Eq. 2.14,

where a_i and l_i are the axial and lateral displacements of the i^{th} sample of that scan line (index j is omitted for brevity) [17], [19]. The difference between echo amplitudes was represented in [19] with the term: $[I_1(i,j) - I_2(i+a_i, j+l_i)]^2$, which replaces the original *sum of absolute differences* term used in [17] with cheaper operations. Temporary axial and lateral displacements $d_a = a_{i-1}$ and $d_l = l_{i-1}$ are used to minimize: $\frac{C_j(d_a,d_l,i-1)+C_{j-1}(d_a,d_l,i)}{2} + R_j(a_i, l_i, d_a, d_l)$, where R_j is a regularization (or smoothness) term defined in Eq. 2.15, that ensures displacement continuity between consecutive samples in the j^{th} scan line; α_a and α_l are axial and lateral regularization weights [17], [19].

$$C_j(a_i, l_i, i) = [I_1(i,j) - I_2(i+a_i, j+l_i)]^2 + \min_{d_a,d_l}\left\{\frac{C_j(d_a,d_l,i-1) + C_{j-1}(d_a,d_l,i)}{2} + R_j(a_i,l_i,d_a,d_l)\right\} \quad [19] \tag{2.14}$$

$$\textit{where} \quad R_j(a_i,l_i,a_{i-1},l_{i-1}) = \alpha_a(a_i - a_{i-1})^2 + \alpha_l(l_i - l_{i-1})^2 \quad [19] \tag{2.15}$$

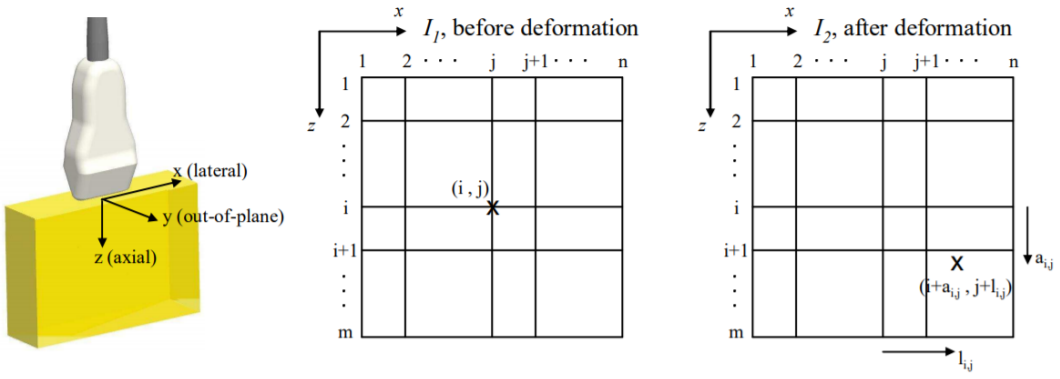


Figure 2.1: Depiction of axial and lateral displacements between I_1 and I_2 [19]

The DP process begins by generating the cost for each sample i in scan line j using C_j, with temporarily assumed displacements $d_a = a_{i-1}$ and $d_l = l_{i-1}$, for $i = 2, 3, ..., m$, where m is the number of samples in a scan line. The last axial a_m and lateral l_m displacements giving minimum C_j at $i = m$ are then used to start the back-propagation on the scan line, where the minimum cost of each sample is used to determine the values for a_i and l_i for the

previous sample, until a_1 and l_1 are obtained [17], [19]. This is repeated for all scan lines. Computation was reduced in [17] by down-sampling data and utilizing a memoization matrix to avoid repeated calculations. To improve robustness against large local decorrelations, one can employ Eq. 2.16, with a weight $w(r_i)$, defined in Eq. 2.17, where r_i is the residual associated with the displacement of the i^{th} sample of the j^{th} scan line. Eq. 2.16 will converge to a local minimum based on the choice of the threshold T in Eq. 2.17 [19].

$$C_j(a_i, l_i, i) = w(r_i)\left[I_1(i, j) - I_2(i + a_i, j + l_i)\right]^2 + \min_{d_a, d_l}\left\{\frac{C_j(d_a, d_l, i-1) + C_{j-1}(d_a, d_l, i)}{2} + R_j(a_i, l_i, d_a, d_l)\right\} \quad [19] \tag{2.16}$$

$$where \;\; w(r_i) = \frac{1}{1 + (\frac{r_i}{T})^2} \quad [19] \tag{2.17}$$

Two-Dimensional Analytic Minimization (2D AM)

Since tissue deformation in ultrasound shear wave elastography is quite small, sub-sampled displacement estimations are required [19]. Two dimensional Analytic Minimization (2D AM) is used to acquire sub-sampled axial $\Delta a_{i,j}$ and lateral $\Delta l_{i,j}$ displacement fields simultaneously to further refine $a_{i,j}$ and $l_{i,j}$ [19]. The process begins by applying DP to a set of *seed* scan lines (typically chosen to be located in the middle of an image) to acquire their integer displacements [19]. Initial sub-sample estimates a_i and l_i for $i = 1, 2, ..., m$ of each *seed* scan line are obtained through linear interpolation of their integer displacements [19]. These initial sub-sample estimates a_i and l_i are then used to acquire Δa_i and Δl_i that minimizes Eq. 2.18 for the *seed* scan lines. The use of Eq. 2.18 is repeated for consecutive scan lines progressing left and right from the *seed* scan lines. Instead of estimating integer displacements and using linear interpolation, the sub-sampled displacement estimates of the previous scan line

are used as the initial sub-sample estimates a_i and l_i for the current scan line.

$$
\begin{aligned}
&C_j(\Delta a_1, \ \ldots\ , \Delta a_m\ , \ \Delta l_1, \ \ldots\ , \Delta l_m) = \\
&\sum_{i=1}^{m}\{[I_1(i,j) - I_2(i + a_i + \Delta a_i, \ \ j + l_i + \Delta l_i)]^2 + \\
&\alpha\,(a_i + \Delta a_i - a_{i-1} - \Delta a_{i-1})^2 + \beta_a\,(l_i + \Delta l_i - l_{i-1} - \Delta l_{i-1})^2 + \\
&\beta_l'\left(l_i + \Delta l_i - l_{i,j-1}\right)^2\} \quad [19]
\end{aligned}
\tag{2.18}
$$

$$
where\ \ \beta_l' = \frac{\beta_l}{1 + |r_{i,j-1}|} \quad [19] \tag{2.19}
$$

The hyper parameters used in Eq. 2.18 are intended to regularize the cost function and support displacement continuity [19]. Specifically, α is used to calibrate the axial proximity of each sample to the previous sample above it, while β_a and β_l' are used to calibrate the lateral proximity of each sample to the previous samples above it (to the left if propagating right, or to the right if propagating left) [19]. Since the sub-sample displacement estimate of each scan line is calculated from the results of the previous scan line, error in the calculation for one scan-line can propagate to all others that follow (though the propagated error decreases exponentially) [19]. To reduce this propagation of error, β_l', defined in Eq. 2.19, is used instead of β_l, where, $r_{i,j-1}$ is the residual associated with the displacement of the i^{th} sample of the previous scan line. The inversely proportional relationship of the residual $r_{i,j-1}$ to β_l' helps to decrease the weight of β_l', if there is a large error in the previous scan line's displacement calculation (reflected in $r_{i,j-1}$). This minimizes displacement underestimation caused by smoothness constraints [19].

2.1.3. Clutter Filter Wave Imaging (CFWI)

Clutter Filter Wave Imaging (CFWI) is an alternative method of detecting mechanical waves without the expensive operations involved in motion estimation techniques [37]. The in-

tention is to apply a simple temporal clutter filter on the IQ data as shown in Figure 2.2, highlighting the attenuated band and making the motion of interest visible. This filter is designed based on the Doppler frequency corresponding to the velocity of the mechanical wave. A first-order Butterworth filter with an appropriate cutoff frequency would suffice for this purpose according to [37]. The cutoff frequency f_c of the filter is determined by the cutoff velocity V_c as shown in Eq. 2.20, 2.21 and 2.22, where c_0 is the speed of sound within the tissue, FPS is the frame rate, f_0 is the transmit frequency, and v_{Nyq} is the Nyquist velocity [37]. The value of V_c is chosen to be within a narrow range of the target mechanical wave velocities in the tissue.

$$f_c = fc_n \times \frac{FPS}{2} \quad [37] \tag{2.20}$$

$$fc_n = \frac{V_c}{v_{Nyq}} \quad [37] \tag{2.21}$$

$$v_{Nyq} = \frac{c_0 \times FPS}{4f_0} \quad [37] \tag{2.22}$$

The resulting filtered IQ signal goes through the typical envelope detection and logarithmic compression, as shown in Figure 2.2. It is then smoothed using a spatio-temporal averaging low-pass filter to improve signal-to-noise ratio [37]. Figure 2.2 shows that differentiating the output of the low-pass filter in time allows the mechanical wave propagation to be seen as bands of attenuated signals moving in the B-mode sequence [37].

CFWI was initially used in RADAR applications to suppress stationary components from received signals in order to isolate moving targets [37]. This technique is primarily used to determine the locality of the tissue velocity induced by the propagating wave [37]. CFWI is more sensitive to subtle tissue displacements compared to motion estimation techniques, therefore, displacement frames produced from this method have values that fluctuate drasti-

cally between neighbouring points, skewing the subsequent speed estimation [37].

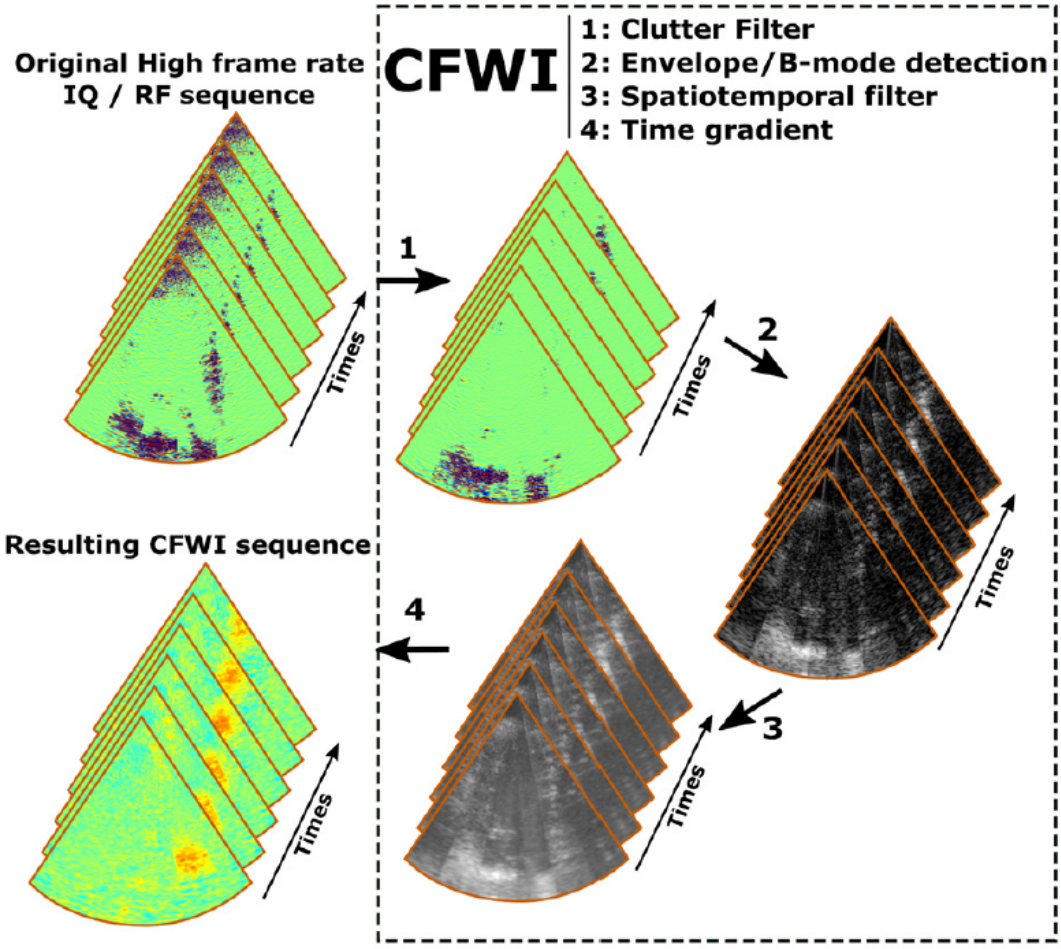


Figure 2.2: Illustration of operations involved in CFWI [37]

2.2. Shear Wave Speed (SWS) Estimation

This section introduces two shear wave speed (SWS) estimation techniques. The second technique is an extension of the first technique, considering a wider range of possible velocity estimates.

2.2.1. Linear Radon Transform

The linear Radon transform (LRT) is the mapping of a recorded IQ signal with spatial and temporal dimensions (x, t) to the Radon domain (ρ, τ) according to Eq. 2.23. A point (ρ, τ) defines a linear trajectory in the (x, t) domain. For the purposes of shear wave speed (SWS) estimation, x is the lateral offset from the origin of the shear wave pulse, t is the elapsed time after the shear wave pulse excitation, τ is the time intercept at zero spatial offset (e.g., center lateral position or origin of the shear wave), and ρ is the slope of time t over spatial position x.

$$m(\rho, \tau) = \sum_{x_{min}}^{x_{max}} d(x, t = \tau + \rho x) \quad [38] \tag{2.23}$$

Using LRT for SWS estimation involves finding the value of ρ that maximizes the sum of displacements $m(\rho, \tau)$. The search for ρ is limited to the range of values that the SWS could be: ρ_{min} to ρ_{max} [38].

2.2.2. Radon Sum Transform

The Radon sum transform (RST) uses the same transformation in Eq. 2.23 to operate in the Radon domain, but it is a more accurate technique than the LRT because it considers more trajectories [39]. Considering a linear speed trajectory spanning from (x_{start}, t_{start}) to (x_{end}, t_{end}), Eq. 2.24 relates the position x to time t. Symbol c given by Eq. 2.25 is the speed along that trajectory, and only trajectories with $x_{end} > x_{start}$ and $t_{end} > t_{start}$ are considered as shown in Figure 2.3, which is based on an example slice (x, t) shown in Fig. 2.4 obtained from an ARFI simulation [39].

$$t = (x - x_{start})\frac{1}{c} + t_{start} \quad [39] \tag{2.24}$$

$$c = \frac{x_{end} - x_{start}}{t_{end} - t_{start}} \quad [39] \tag{2.25}$$

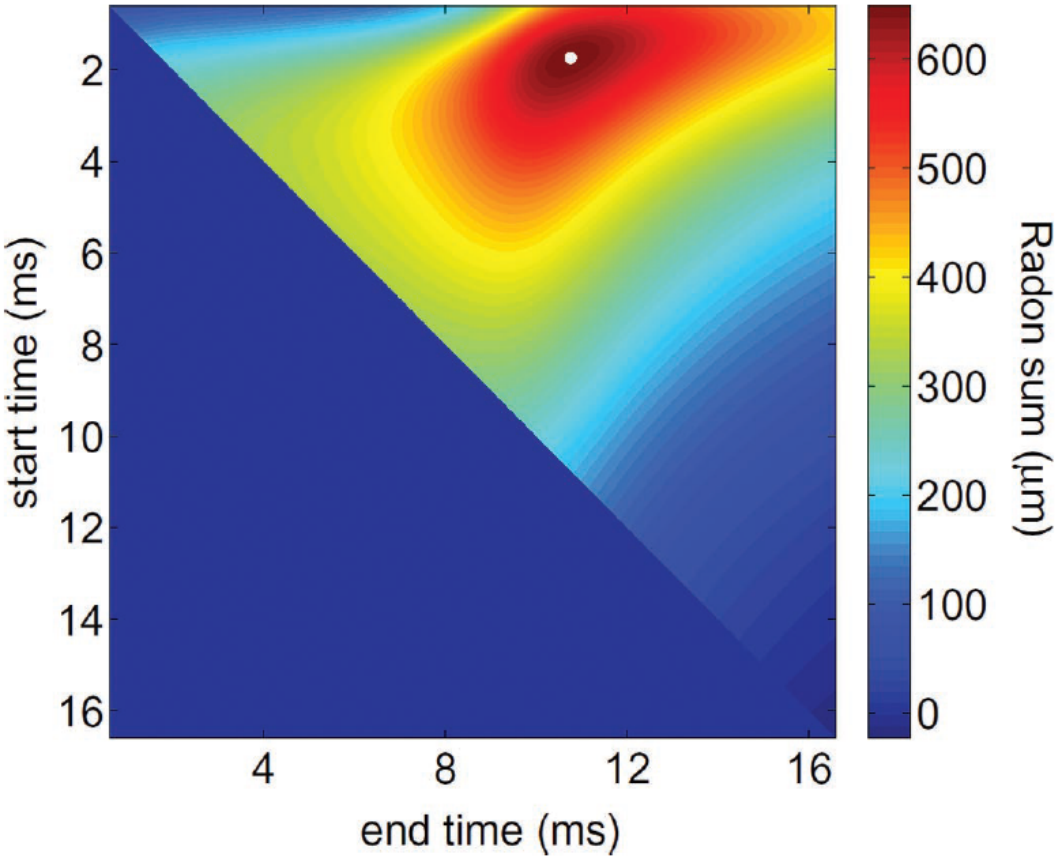


Figure 2.3: Radon sum transformation of a slice (x, t) of an ARFI simulation [39]

The Radon sum transform shown in Figure 2.3 was computed according to Eq. 2.26, where t is obtained from Eq. 2.24. This is similar to the LRT given by Eq. 2.23, where $\rho = \frac{1}{c}$ and $\tau = t_{start}$. However, the difference is that in the Radon sum transform, x_{start} can be at a position other than zero (origin of the ARF pulse). This allows for better fitted trajectories to be taken into account when searching for the largest sum of displacements [39].

$$S(t_{start}, t_{end}) = \sum_{x_{start}}^{x_{end}} d(x, t) \quad [39] \tag{2.26}$$

Figure 2.4 highlights the selected trajectory based on the largest output of the RST of a given slice (x, t). In comparison to the LRT, the RST is better suited for cases where the propagating pulse trajectory does not cross the origin of the pulse, which is a common scenario [39].

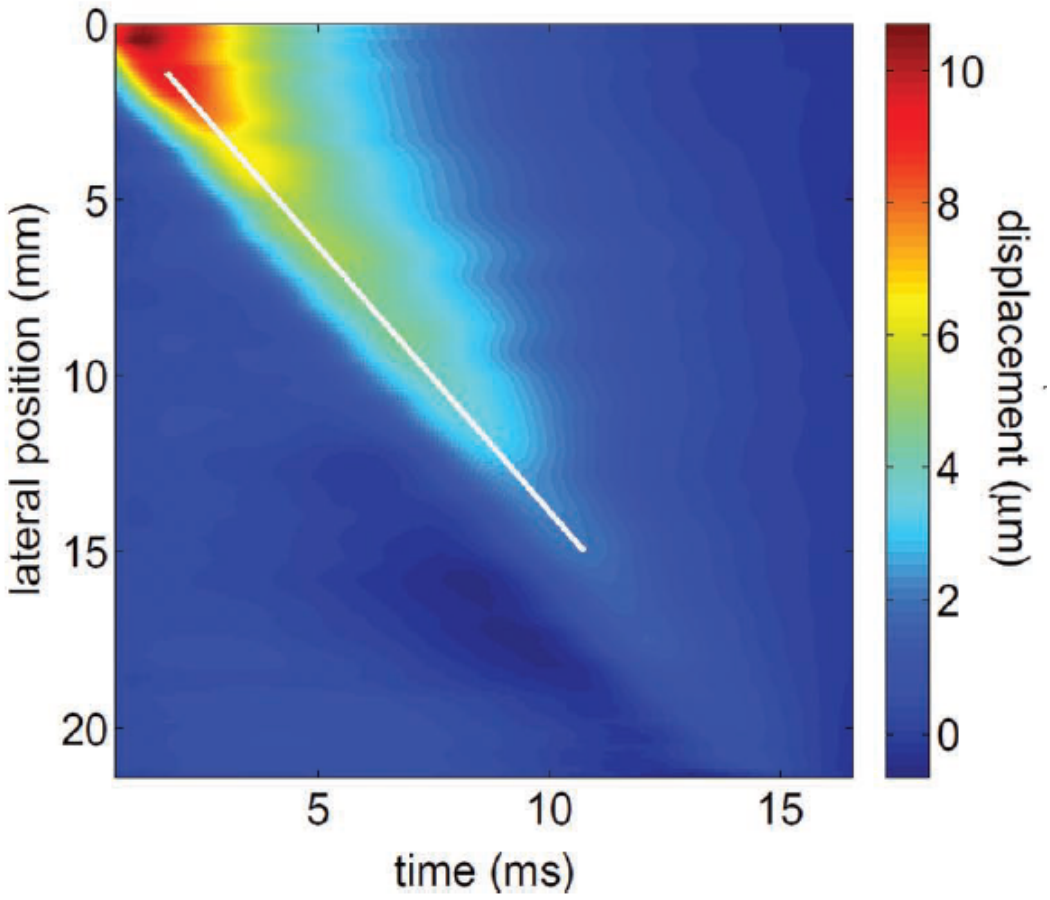


Figure 2.4: Depiction of the slice used in Figure 2.3 with the optimal shear wave trajectory (white line) based on the largest Radon sum [39]

2.3. Our Approach

We applied various combinations of the displacement and SWS estimation techniques to the public-domain ARFI dataset provided by the USTB team [20], [40]. It was acquired from a CIRS Elasticity QA Spherical Phantom, which contained eight spheres as shown in Figure 2.5, each color corresponding to a lesion type listed in Table 2.1 [41]. The larger spheres have

a diameter of 20 mm and the smaller ones have a diameter of 10 mm. The dataset in question contained raw RF channel records corresponding to a phantom section, containing the Type IV 10 mm sphere (the smaller red sphere) at 15 mm depth. These records were acquired using a Verasonics research scanner with a 128-element L7–4 linear array probe, whose parameters are specified in Table 2.2 [40]. The raw RF data consist of four super-frames (four groups of frames corresponding to four separate and non-overlapping ARF pulse transmit events). Each super-frame contains 50 frames captured using high frame rate plane wave imaging [20], [40]. The approximate location of the Type IV sphere with respect to the origin of the AFR pulse is x = 10 mm and z = 15mm [20], [40].

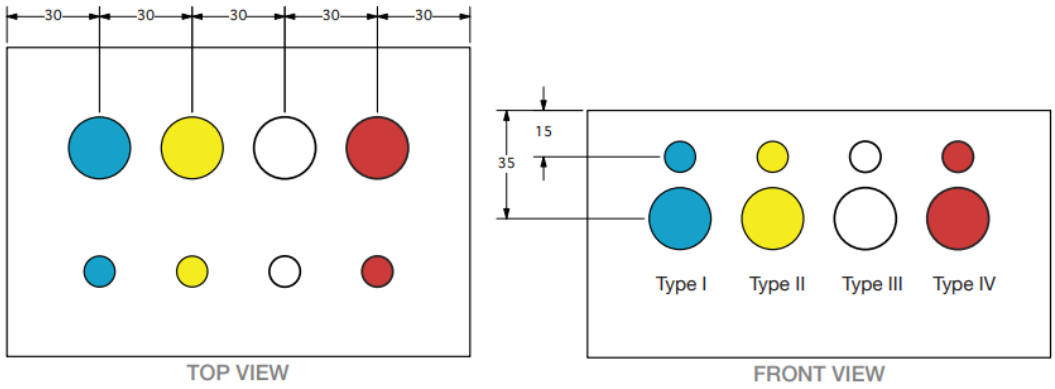


Figure 2.5: View of the eight spheres in the CIRS Elasticity QA Spherical Phantom

The USTB organizes its channel data into a four-dimensional matrix of size $Ns \times Nc \times Nw \times Nf$, where Ns is the number of samples obtained per channel, Nc is the number of channels or sensing elements, Nw is the ultrasound transmissions per frame (this would be a single transmission in our case), and Nf is the number of frames [20], [21]. The dataset in question has the size of $1664 \times 128 \times 1 \times 200$, and it is stored in a Universal File Format (UFF) developed by Structural Dynamics Research Co. (SDRC) [20]. This format was inspired by a demand for repeatability of experiments between Computer Aided Design (CAD) and Computer Aided Test (CAT) software packages with the goal of disseminating results [20], [40].

Table 2.1: Lesion types in CIRS Elasticity QA Spherical Phantom

Lesion Type	Young's Modulus
Type I	8 kPa
Type II	14 kPa
Type III	45 kPa
Type IV	80 kPa
Background	25 kPa

Table 2.2: Acquisition parameters: Verasonics research scanner with 128-element L7–4 linear array probe

Ultrasound Scanner Parameters	Setting
Pitch	0.2980 mm
Element Width	0.2500 mm
Element Height	3.0000 mm
Number of Elements	128 elements
Speed of Sound in Tissue	1540 m/s
Scanner Sampling Frequency	20.833 MHz
ARFI Pulse Center Frequency	5.208 MHz
Apodization f_number	1.7
Apodization Hanning Window	5 samples
Minimum Aperture	1.000×10^{-3} mm
Maximum Aperture	10 mm

The USTB contains a linear scanning function that can be used to divide the channel data into synthetic scan lines. Two scan-lines per sensing element was the recommended setting

for this dataset, resulting in 256 synthetic scan-lines [20]. The corresponding beamformed IQ dataset (used for subsequent displacement and SWS estimation) was obtained using a conventional delay-and-sum (DAS) technique [42]. A baseline implementation of this DAS beamformer is provided by the USTB and was used in this work without modifications.

From the displacement and SWS estimation techniques described in this chapter, this report will show that the Doppler frequency estimation based on auto-correlation used in combination with the Radon sum transform yields better overall speed estimates compared to other combinations for both the interior of the Type IV inclusion and outside of it (background).

It should be noted that during the course of this project we also experimented with other techniques. Specifically, we explored two alternative displacement estimators based on GLUE [18] and CGLUE [4] methods, as well as three alternative SWS estimators based on linear regression [43], Random Sample Consensus (RANSAC) [44], and Fourier-domain shift matching (FDSM) [45]. However, our application of these techniques to the ARFI dataset under consideration did not produce acceptable speed estimates.

This report not only presents a comprehensive comparison of different combinations of displacement and SWS estimation techniques, but also highlights the benefits of using directional filtering and a Canny edge detection mask to process noisy displacement data. Directional filtering [46] reduces in-plane reflected waves, while Canny edge detection [47] helps highlight the most prominent trajectories used by the Radon sum transform to extract speed estimates. The latter are also subjected to morphological opening operations to promote continuity of the estimated SWS profiles. Further details are provided in the next chapter.

3. PROPOSED METHOD AND IMPLEMENTATION

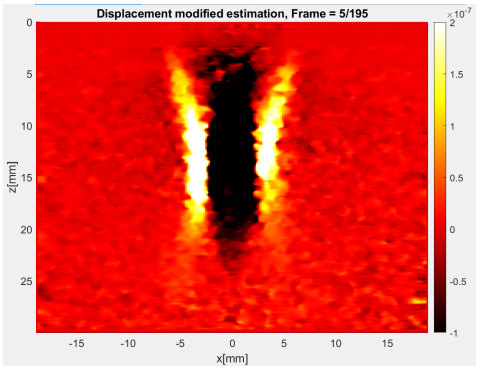
This chapter provides details on the implementation of our approach discussed in Section 2.3. It further elaborates on techniques for post-processing displacement frames and shear wave speed (SWS) images.

3.1. Displacement Data Processing

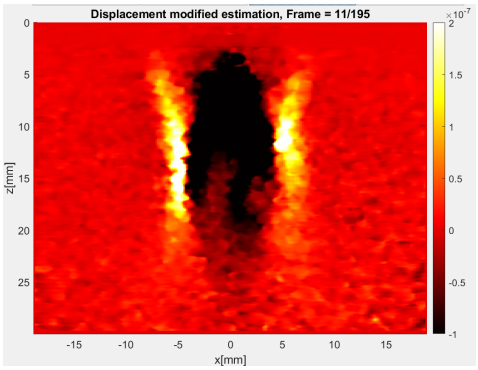
Given the beamformed IQ data, the displacement estimation method, *modified _autocorrelation_displacement_estimation*, was used to acquire displacement frames. This USTB-provided method performed the Doppler frequency displacement estimation technique based on auto-correlation introduced in [22]. The collection of displacement frames form a three-dimensional data set $D(z, x, t)$, where z is the depth along beam lines, x is the lateral position of a sensing element, and t identifies a frame. This dataset is then divided along the z axis into two-dimensional (x, t) slices, each corresponding to a specific depth value of interest. Each slice is further split into two parts along the t axis. The right-side part starts at the origin of the ARF pulse and includes (x, t) displacement data capturing the shear wave propagation to the right of the pulse. Similarly, the left-side part includes the rest of (x, t) displacement data capturing the shear wave propagation to the left of the pulse. A directional filter was applied to each part of a slice to help reduce artifacts caused by reflected waves traveling in the opposite direction of the expected shear wave propagation. An edge isolating mask was also applied to each slice to highlight trajectories, and the Radon sum transform was used to estimate SWS within small lateral windows sliding along the x axis. The benefits of directional filtering and edge masking are illustrated in the next two sections.

3.1.1. Directional Filtering

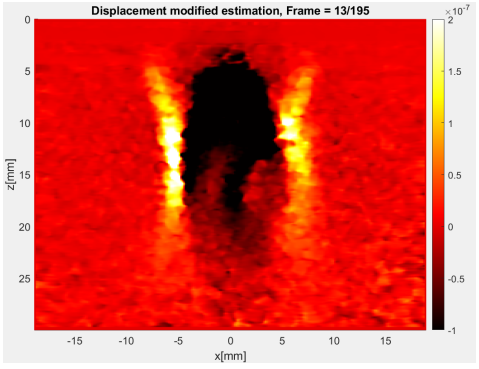
A simple two-dimensional directional filter was used on each slice of the displacement data to help reduce in-plane reflected waves [46], [48]. Figure 3.1 shows an example of such waves present in displacement data. This filter is implemented in the Fourier domain (k_x, ω) corresponding to the spatio-temporal dimension (x, t) indicating the position of sensing elements x and frame number t [46]. The filter aids in separating leftward and rightward propagating waves, and waves from multiple simultaneous steered excitations [46], [49], [50]. The filter defines responses to variations in spatial frequencies k_x and temporal frequencies ω such that waves moving away from the origin of the ARF pulse are passed, and those traveling in the opposite direction are attenuated [46], [48].



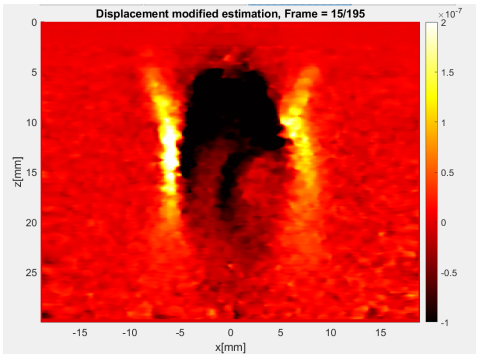
(a) Frame 5



(b) Frame 11



(c) Frame 13



(d) Frame 15

Figure 3.1: Depiction of in-plane reflected waves in displacement data

Notice the ripple on the right half of the image travelling leftward against the shear wave.

Figure 3.2 shows a depiction of the filter in the Fourier domain. To reduce a ringing artifact resulting from a drastic change in filter response, the work in [46] suggest setting the quadrant boundaries to 0.5. The left and right halves of each slice of the displacement data was processed separately, as shown on lines **11.** and **13.** of the pseudo-code in Figure 3.3. Line **12.** of the pseudo-code shows that the left side was flipped on its t axis, to be processed similarly to its right counterpart. Lines **15.** to **18.** shows that each side of a slice was Fourier transformed, and then multiplied element-wise by the directional filter shown in Figure 3.2. The inverse Fourier transform was used to recover the filtered slice in its spatio-temporal domain (x, t).

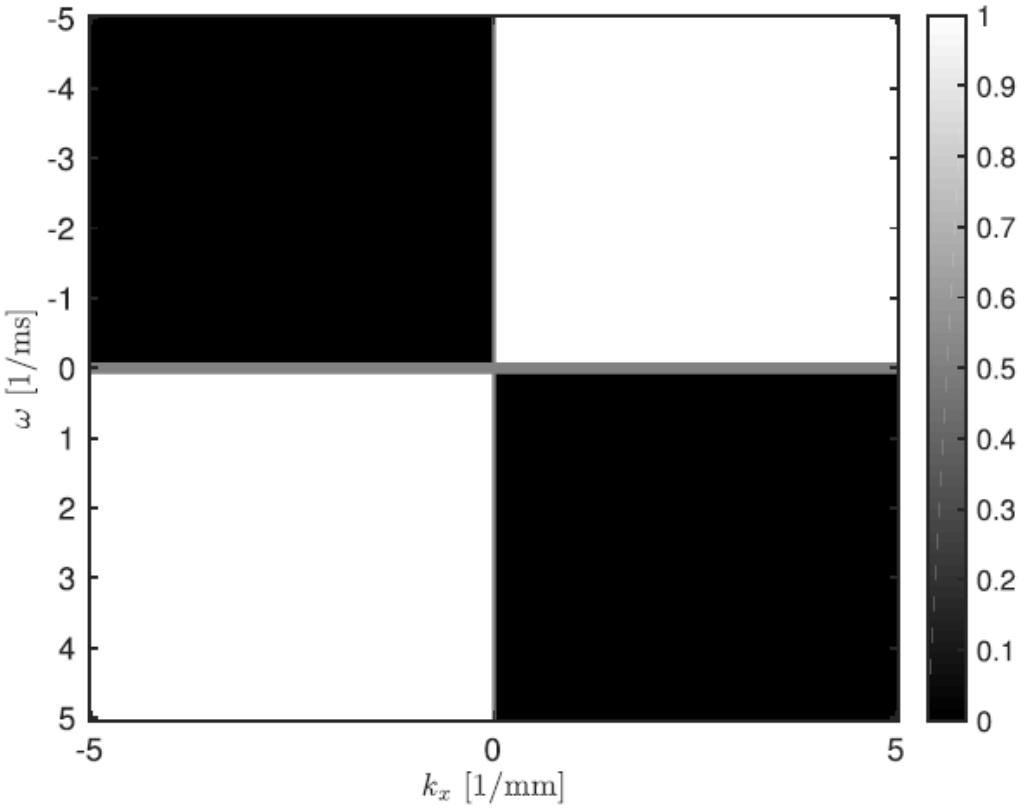


Figure 3.2: Two-dimensional Spatio-temporal Fourier Domain Directional Filter [46]

```
displacement_data = modified_autocorrelation_displacement_estimation(beamformed_data)
[Nz, Nx, Nt] = size(displacement_data)
Nx_FFT = round_up_to_closest_power_of_2(Nx/2)
Nt_FFT = round_up_to_closest_power_of_2(Nt)
% Create directional filter
directional_filter = matrix_of_ones(Nx_FFT, Nt_FFT) % size [Nx/2, Nt]
directional_filter[1 to Nx_FFT/2, 1 to Nt_FFT/2] = 0
directional_filter[Nx_FFT/2+1 to Nx_FFT, Nt_FFT/2+1 to Nt_FFT] = 0
for z = 1 to Nz :
        slice = displacement_data[z, all, all] % Each slice is of size [Nx, Nt]
        left_half = slice[1 to Nx/2, all] % size [Nx/2, Nt]
        left_half = flip_x_axis(left_half)
        right_half = slice[(Nx/2)+1 to Nx, all] % size [Nx/2, Nt]
        % Apply directional filter
        left_half_fft = fft(left_half, size=[Nx_FFT, Nt_FFT])
        left_half = inverse_fft(left_half_fft × directional_filter, size=[Nx/2, Nt])
        right_half_fft = fft(right_half, size=[Nx_FFT, Nt_FFT])
        right_half = inverse_fft(right_half_fft × directional_filter, size=[Nx/2, Nt])
⋮
```

Figure 3.3: Pseudo-code Portion Related to Directional Filtering

Figures 3.4 and 3.5 show an example of applying directional filtering to the right-side part of a (x, t) slice. The long arrows indicate the expected pulse trajectory (i.e., x increases as t increases). The short arrow in Figure 3.4 identifies a reflected wave direction. The latter is no longer present in Figure 3.5 after filtering.

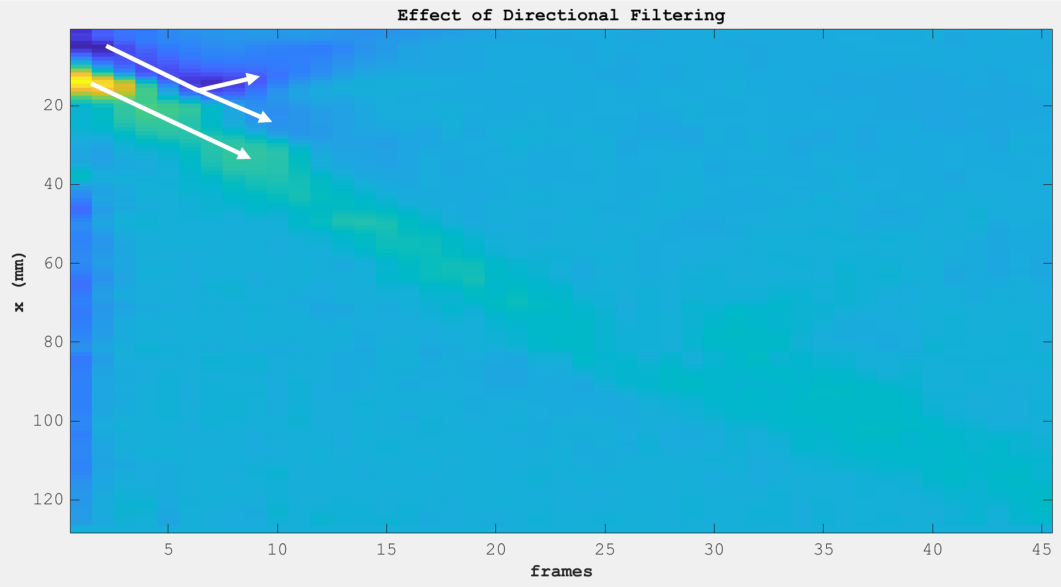


Figure 3.4: Effect of Directional Filtering: Original Image Marked

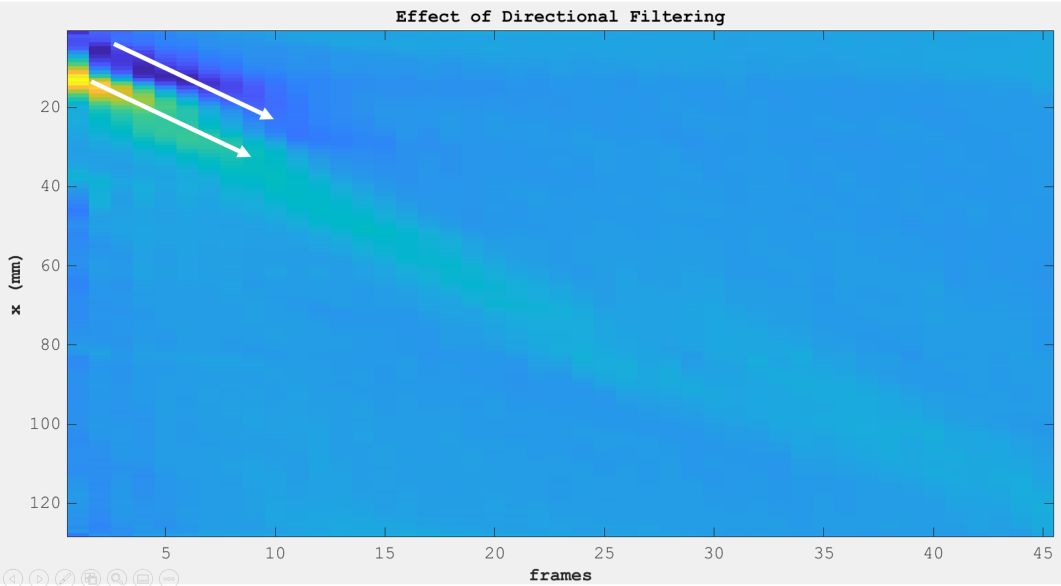


Figure 3.5: Effect of Directional Filtering: Filtered Image Marked

3.1.2. Canny Edge Detection Mask

Extracting edges is an effective method of highlighting wave trajectories from the background. Such trajectories can be identified by rapid change from low displacement values to high displacement values and vice versa. This rapid change corresponds to high spatial

frequencies representing edges. The Canny edge detection algorithm was chosen to extract edges due to its ability to find all *true* edges (edges that are preserved after non-maximum suppression) [47]. Non-maximum suppression is an attractive feature of the Canny edge detection algorithm because it results in a single edge point response, which provides SWS estimators with a clearly defined trajectory [47]. The Canny edge detection algorithm's ability to localize edges and link correlated edges also simplifies the task of an SWS estimator [47].

The objective of using Canny edge detection is to obtain well defined curves that capture shear wave propagation. MATLAB's implementation of the Canny edge detection algorithm *edge(I, 'Canny')* was used to create a binary mask from each slice that was then applied to that slice as shown on lines **20.** and **21.** of the pseudo-code in Figure 3.6. Figures 3.7 and 3.8 shows half of a slice (x, t) before and after edge-mask filtering, respectively, with a Canny edge detection mask made from that slice. It is evident that a slope of a trajectory would be easier to extract from Figure 3.8 compared to Figure 3.7, producing a more reliable SWS estimate. This is one novel aspect of the project in relation to other works on shear wave elastography.

```
displacement_data = modified_autocorrelation_displacement_estimation(channel_data)
[Nz, Nx, Nt] = size(displacement_data)
⋮
for z = 1 to Nz :
        slice = displacement_data[z, all, all] % Each slice is of size [Nx, Nt]
⋮
        % Apply directional filter
⋮
        % Apply Canny edge detection mask
        left_half = abs(left_half) × edge(abs(left_half),'Canny')
        right_half = abs(right_half) × edge(abs(right_half),'Canny')
⋮
```

Figure 3.6: Pseudo-code Portion Related to Canny Edge Detection Masking

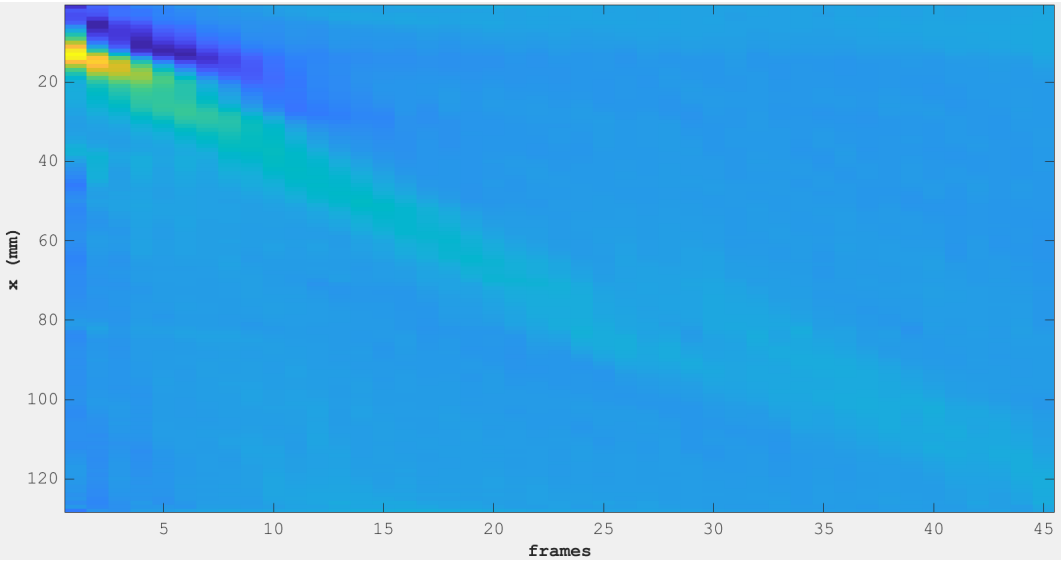


Figure 3.7: Effect of Canny Edge Detection Masking: Original Image

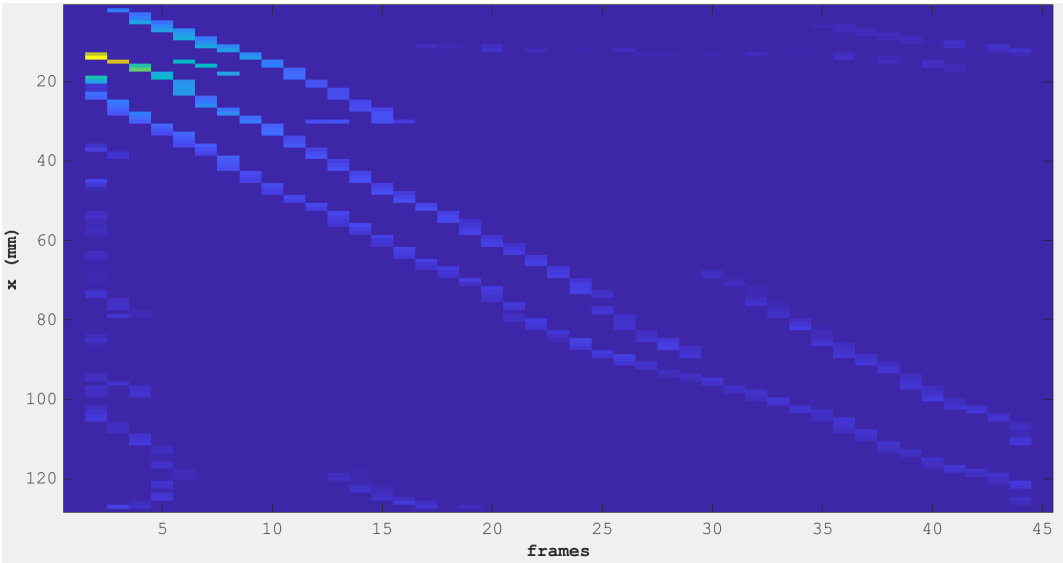


Figure 3.8: Effect of Canny Edge Detection Masking: Filtered Image

3.1.3. Sliding Window Radon Sum Transform

We used a sliding window along the x axis to determine a single SWS value for the center of that window. The SWS value in question was obtained using the Radon sum transform described in Section 2.2.2. The window length was a user-defined parameter. Lines **23.** to **26.** of the pseudo-code in Figure 3.9 show how the sliding window was implemented for both right-side and left-side parts of each (x, t) slice.

MATLAB provides a built-in function *radon(I)* that performs the Radon transform in the form shown in Eq. 3.1. This equation is equivalent to Eq. 2.24 from Section 2.2.2 of Chapter 2, but it operates with different parameters [51]. Eq. 2.24 uses the slope ρ of time t over spatial position x, and the time intercept τ at zero spatial offset, whereas, Eq. 3.1 uses the angle θ above the spatial x axis and the radius r from the origin of the (x, t) domain [51]. The tutorial [51] provides a detailed proof of the equivalence of Eq. 2.24 and Eq. 3.1.

$$m(r,\theta) = \sum_{t_{min}}^{t_{max}} \sum_{x_{min}}^{x_{max}} d(x,t)\ \ \delta(r - x\ cos\theta - t\ sin\theta)\ \ dx\ dt \quad [51] \tag{3.1}$$

Using MATLAB's *radon(I)* function, the SWS can be computed as shown on lines **28.** to **31.** of the pseudo-code in Figure 3.9. The outputs of the Radon transform undergoes partial-differentiation $\frac{\partial(m(r,\theta))}{\partial\theta}$, and the absolute value of the difference is summed along the r axis, resulting in the angle corresponding to the largest accumulation of r values. The arc-tangent of that angle produces the estimated speed of the most prominent trajectory in the (x,t) domain. Lines **30.** to **32.** of the pseudo-code in Figure 3.9 shows how the desired SWS estimates are computed for each x within each (x,t) slice for each depth z.

```
1.  displacement_data = modified_autocorrelation_displacement_estimation(channel_data)
2.  [Nz, Nx, Nt] = size(displacement_data)
    ⋮
9.  for z = 1 to Nz :
10.         slice = displacement_data[z, all, all] % Each slice is of size [Nx, Nt]
    ⋮
14.         % Apply directional filter
    ⋮
19.         % Apply Canny edge detection mask
    ⋮
22.         % Sliding window
23.         for x = 1 to (Nx/2)-(window_length/2)+1 :
24.                 window = x to minimum_between(x+window_length-1, Nx/2)
25.                 left_half_windowed = left_half[window, all] % size [window, Nt]
26.                 right_half_windowed = right_half[window, all] % size [window, Nt]
27.                 % Implementation of the Radon Sum Transform
28.                 angle_left = sum(abs(difference(radon(left_half_windowed))))
29.                 angle_right = sum(abs(difference(radon(right_half_windowed))))
30.                 sws_left[window, all] = tan_inverse(angle_left)
31.                 sws_right[window, all] = tan_inverse(angle_right)
32.         sws = [flip_x_axis(sws_left), sws_right] % size [Nx, Nt]
```

Figure 3.9: Pseudo-code Portion Related to Sliding Window Radon Sum Transform

3.2. Morphological Filtering of SWS Estimates

From experimentation, it was found that a noisy dataset could yield SWS images that contain outliers and irregular speed estimates. To help improve the SWS images acquired in Section 3.1.3, morphological operations were performed on them. The advantage of using morphological operations over standard noise reduction algorithms is that morphological operations retain structural information [52].

Morphological operations involve using a structuring element (SE) to transform an image through neighborhood operations [52]. An SE is defined according to a specific geometric shape, size and number of intensity levels. This is determined with prior knowledge of the image based on expected characteristic patterns [52]. Choosing an SE with a larger size than the noise pattern results in reducing noise when opening or closing morphological operations are applied [52].

The opening operation $\circ$ is defined according to Eq. 3.4, where f is the noisy SWS image with domain F, and g is the selected SE with domain G. This operation utilizes the erosion operation $\ominus$ and the dilation operation $\oplus$ as defined by Eq. 3.2 and Eq. 3.3, respectively. Eq. 3.5 defines the closing operation, an alternative to the opening operation.

$$Erosion \;:\quad (f \ominus g)(X) = \min_{Z \in G \;\&\; X+Z \in F} f(X+Z) - g(Z) \quad [52] \tag{3.2}$$

$$Dilation \;:\quad (f \oplus g)(X) = \max_{Z \in G \;\&\; X-Z \in F} f(X-Z) + g(Z) \quad [52] \tag{3.3}$$

$$Opening \;:\quad f \circ g = (f \ominus g) \oplus g \quad [52] \tag{3.4}$$

$$Closing \;:\quad f \bullet g = (f \oplus g) \ominus g \quad [52] \tag{3.5}$$

MATLAB provides functions that implement these morphological operations. Since eliminating small features in the noisy SWS image is desirable, the opening operation implemented in MATLAB's *imopen* function was used to achieve a blurring effect while emphasizing contrast. A flat/binary disk structuring element was chosen to avoid introducing orientational and directional biases within the AOI. The radius of the SE is a user-defined parameter promoting SWS estimate continuity (smoothing).

4. EVALUATION RESULTS

This chapter summarizes the experimental setup of each displacement and shear wave speed (SWS) estimation technique described in Chapters 2 and 3. It also reports the performance of these technique and highlights their benefits and drawbacks.

4.1. Evaluation Setup

In this section, the effects of setup parameters were investigated. Details on the experiments that produced better results compared to others are discussed.

4.1.1. Setup Parameters for Doppler Frequency Estimator in the USTB

As mentioned in Section 2.1.1, USTB provides displacement estimation functions derived from those used for blood velocity measurements. Section 3.1 noted that the *modified_auto-correlation _displacement_estimation* function was chosen amongst other functions in the tool-box due to the visual improvement in displacement frames observed. The stride along the x axis during the auto-correlation was set to 4 samples. This enabled us to compare two sensing elements at a time after the up-scaling by the factor of two along the x axis performed by the DAS beamformer in Section 2.1.1. In other words, two samples originated from a single sensor.

The number of points associated with the z axis from the DAS beamformer was 1024 points. The stride along the z axis was set to 16 samples. Although the frames are formed as square matrices of pixels, the physical values of the step sizes Δz and Δx between pixels would be taken into account during displacement estimation. Specifically, we had Δz and Δx equal to 3.9113×10^{-2} mm and 1.4842×10^{-1} mm, respectively.

As mentioned in Section 3.1.3, the sliding window length is the number of positions along the x axis considered in the calculation of a single SWS value. This sliding window moved with a stride of one position per iteration along the x axis. Since the section of the CIRS Elasticity QA Spherical Phantom under consideration contains a Type IV 10 mm sphere, the window length was set to 3 mm to be sufficiently small to capture the structure of the sphere. Experimentation showed that a significant reduction in the window length resulted in a drastic increase in the noise (invalid SWS estimates that were outside the expected range of velocities), and a significant increase in the window length resulted in loss of resolution.

4.1.2. Setup Parameters for 2D AM

In Section 2.1.2 of Chapter 2, the Analytic Minimization (AM) algorithm was introduced. An optimization-based displacement estimator derived from this algorithm was implemented by the authors of [19] and adopted as part of the combinations of techniques investigated in this project. Section 2.1.2 provided a detailed explanation of the two-dimensional AM (2D AM) algorithm to acquire both axial and lateral sub-samples from integer displacement frames. These integer displacement frames were initially obtained through the Dynamic Programming (DP) process described in Section 2.1.2. The parameters associated to the regularization term R_j, added to the cost function to be optimized, were the axial α_a and lateral α_l regularization weights [19]. Both regularization terms were set to 0.15, which was the recommended value in [19]. The weight $w(r_i)$ for the difference between echo amplitudes, mentioned in Section 2.1.2 to achieve robustness towards large local decorrelations, was dynamically set for each sample on every scan line by specifying its convergence threshold T = 0.4, which was the recommended value in [19]. The range of variations in axial and lateral displacements considered by the optimization-based algorithm was set to $+/-5$ pixels. This was chosen based on the largest reasonable displacement expected from the CIRS Spherical Phantom dataset based on other displacement estimation algorithms.

Other parameters of the cost function for the 2D AM algorithm include the axial and lateral calibration parameters. The axial calibration parameter α was used to adjust the axial proximity of each sample to the previous sample above it [19]. The lateral calibration parameters β_a and β_l were used to adjust the lateral proximity of each sample to the previous sample above it and laterally preceding it (in the direction of the propagating wave), respectively [19]. The parameters α and β_l were set to their recommended values of 5 and 0.005, respectively [19]. The recommended value for β_a was 10 [19]. However, due to the noisy nature of the CIRS Spherical Phantom dataset, the value of β_a used was 20 to allow for more aggressive regularization of the cost function to give greater weight to similar patterns.

4.1.3. Setup Parameters for CFWI

As discussed in Section 2.1.3, the Clutter Filter Wave Imaging (CFWI) technique involves applying a temporal band-pass filter to the IQ data obtained from beamforming the CIRS Spherical Phantom dataset. This results in highlighting low frequency bands by attenuating frequencies related to velocities well above the expected maximum and attenuating static frequencies [37]. The cutoff velocity parameter V_c in the CFWI technique in Section 2.1.3 was set to 4 m/s. This value was used because the lesion type under investigation is a Type IV with a Young's Modulus E of 80 kPa. Assuming that the material density ρ is 1000 kg/m^3, applying Eq. 4.1 and Eq. 4.2 yields the wave velocity v through the lesion of approximately 5.16 m/s. The wave velocity outside the lesion is approximately 2.89 m/s, associated with the background Young's modulus of 25 kPa. The work in [53] used the same CIRS phantom as this project and obtained nominal SWS values of 4.81 ± 0.49 m/s within the lesion, and 2.83 ± 0.38 m/s in the background from the manufacturers. Starting from these expected speeds and adjusting based on experimentation, V_c = 4 m/s resulted in sufficient attenuation of the high frequency bands and static frequencies. Regarding other parameters for the CFWI technique, the speed of sound c_0 in tissue was 1540 m/s, the frame rate was assumed to be

11000 frames/sec, and the center frequency f_0 was 5 MHz.

$$E = 3 \times \mu \quad [54] \tag{4.1}$$

$$v = \sqrt{\frac{E}{3 \times \rho}} \quad [53], [54] \tag{4.2}$$

4.1.4. Experimental Cases Compared

In this project, several combinations of displacement estimation techniques and SWS estimation techniques were explored. Upon preliminary experimentation on the SWS estimation techniques, the Radon sum transform (RST) discussed in Section 2.2 produced SWS images that were closer to the expected values and positions of velocities compared to techniques such as linear regression, RANSAC and other time-of-flight (TOF) speed estimation techniques. Although the performance of the Linear Radon Transform (LRT) was insufficient, it is worth mentioning that it produced better estimates than the other TOF techniques for this dataset, and it served as the foundation for the Radon Sum Transform. The Radon Sum Transform was combined with the displacement estimation techniques discussed in Section 2.1.

This chapter focuses on three cases that yielded results that were closest to the ideal shear wave image. Details of these cases are summarized in Table 4.1. Morphological operations described in Section 3.2 were performed on each case to reduce the effects of irregular speed estimates and noise. The Canny Edge Detector Mask discussed in Section 3.1.2, a unique aspect of the project compared to other works, was used in all cases to support SWS estimators. Case 3 is the recommended combination of techniques that yields the closest values of velocities within the lesion and in the background. The results of the cases in Table 4.1 are presented in further sections of this chapter.

Table 4.1: List of post-beamforming data processing procedures detailed in this report

Procedure	Details
Case 1 :	• An implementation of the CFWI [37] • Radon Sum Transform SWS estimation [39] • Directional Filtering [46] • Canny Edge Detection Mask [47] • Disk structuring element (Morphological operation) [52]
Case 2 :	• The implementation of the 2D AM provided by the authors of [19] • Radon Sum Transform SWS estimation [39] • Directional Filtering [46] • Canny Edge Detection Mask [47] • Disk structuring element (Morphological operation) [52]
Case 3 :	• The USTB implementation of the Doppler frequency displacement estimation based on auto-correlation [20], [22] • Radon Sum Transform SWS estimation [39] • Directional Filtering [46] • Canny Edge Detection Mask [47] • Disk structuring element (Morphological operation) [52]

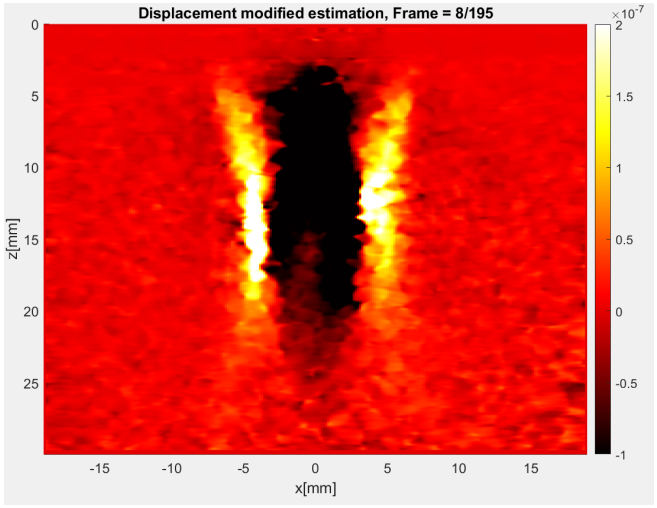
4.2. Displacement Images

This section presents the results of applying the displacement estimation techniques discussed in Section 4.1 to the CIRS Spherical Phantom dataset. The resulting displacement frames from each technique are then visually analysed and linked to the attributes of their respective techniques. These techniques are also compared based on the run time measurements made using MATLAB's code profiling tool [55]. This tool can be used to discover potential improvements to be made based on identifying which functions required the most time and comparing them to the overall execution time [55].

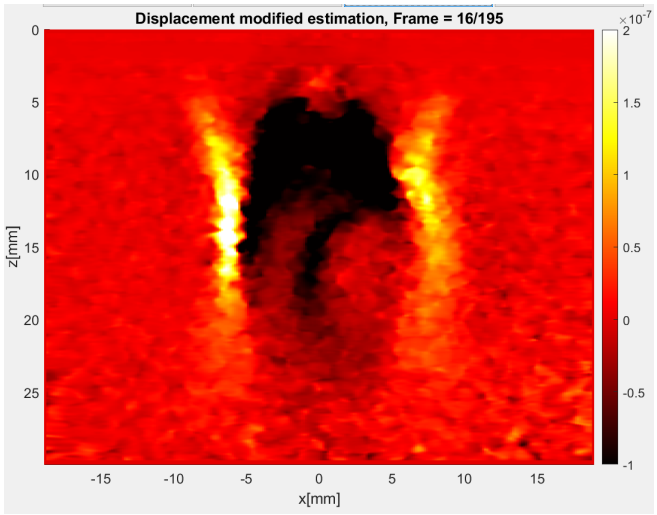
4.2.1. Performance of Doppler Frequency Estimator Based on Auto-correlation

Figure 4.1 shows displacement frames produced by applying the Doppler frequency estimation based on auto-correlation to the CIRS Spherical Phantom dataset. The Doppler frequency estimator is capable of extracting a mean velocity in small regions as discussed in Section 2.1.1. Figure 4.1 shows a few frames evenly spaced to depict the shear wave motion after the ARF pulse. Notice that the faster right-side wave-front propagation is evident between Frames 16 and 24 in Figure 4.1b and Figure 4.1c, respectively, and between Frames 24 and 32 in Figure 4.1c and Figure 4.1d. This is because the wave travelled through the lesion between these frames.

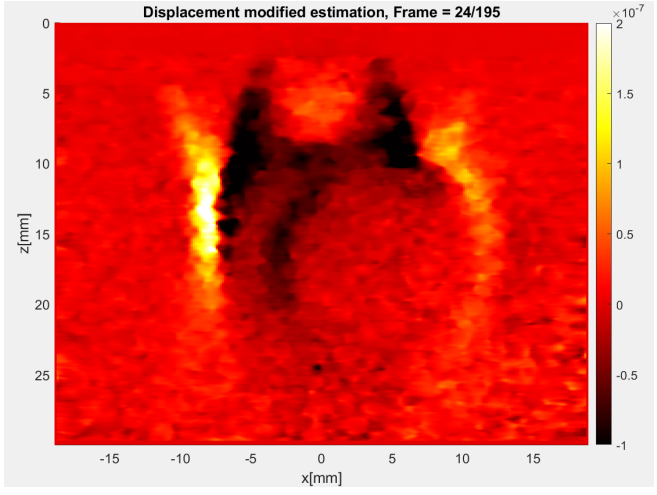
Section 2.1.1 showed that this technique involves some computationally complex operations, such as computing auto-correlation $R(T)$. This results in computation time to be relatively higher than the other two displacement estimation techniques considered in this chapter. As shown in Figure 4.2, boxed in red, this Doppler frequency estimation technique had a run time of 11.540 seconds when applied to the CIRS Spherical Phantom dataset.



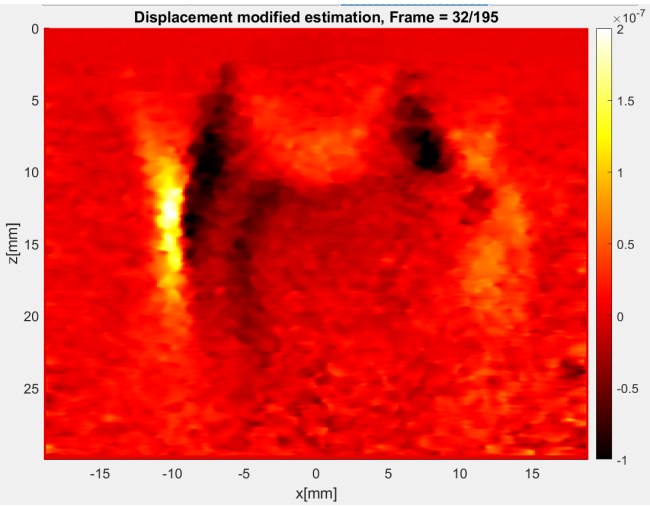
(a) Frame 8



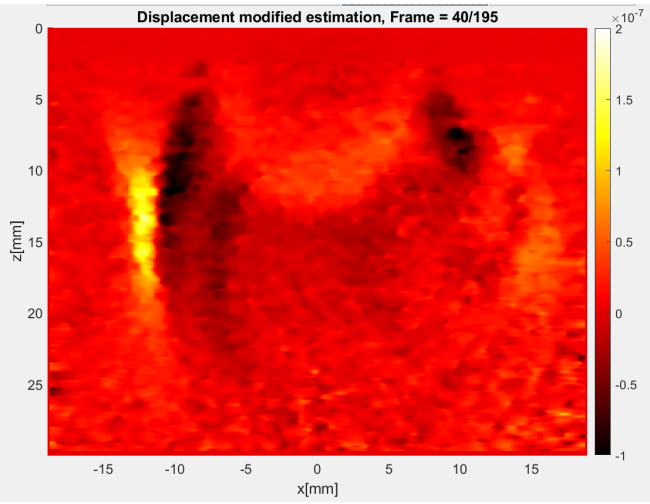
(b) Frame 16



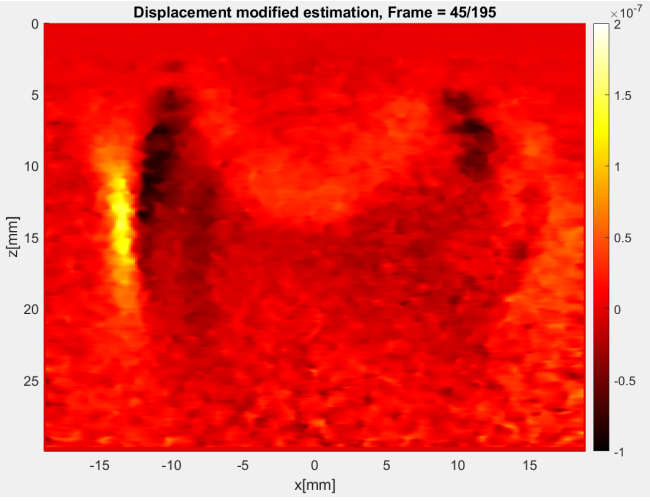
(c) Frame 24



(d) Frame 32



(e) Frame 40



(f) Frame 45

Figure 4.1: Displacement frames obtained using Doppler frequency displacement estimation based on auto-correlation

Profile Summary

Generated 06-Nov-2021 15:57:17 using performance time.

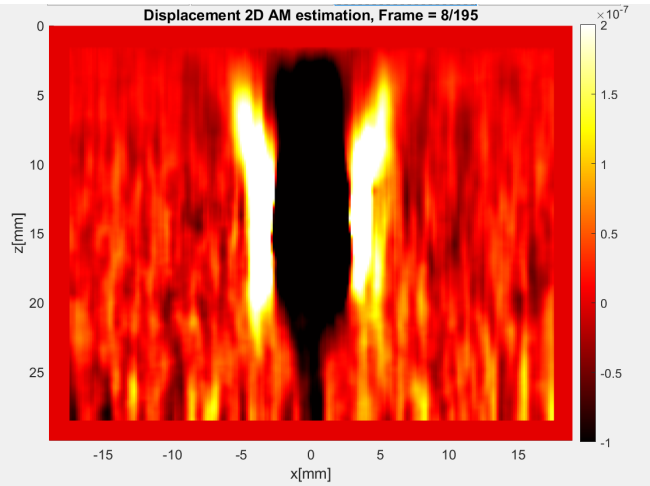
Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
...orrelation_displacement_estimation.go	1	28.457 s	1.566 s	
workbar	197	15.329 s	8.465 s	
...ulsed_doppler_displacement_estimation	195	11.540 s	11.085 s	
...asPlugin>CanvasPlugin.createCanvas	1	3.880 s	2.715 s	
imshow	1	1.869 s	0.637 s	
...tup>CanvasSetup.createScribeLayers	1	0.939 s	0.243 s	
...vasPlugin>CanvasPlugin.getInstance	1	0.757 s	0.757 s	
...anager>ScribeStackManager.getLayer	5	0.698 s	0.444 s	
newplot	2	0.488 s	0.063 s	
angle	390	0.455 s	0.455 s	
newplot>ObserveAxesNextPlot	2	0.407 s	0.020 s	
cla	2	0.386 s	0.070 s	
basicImageDisplay	1	0.309 s	0.303 s	
graphics\private\claNotify	2	0.269 s	0.269 s	
...ger>ScribeStackManager.createLayer	3	0.237 s	0.125 s	
imageDisplayParseInputs	1	0.237 s	0.075 s	
workbar>sec2timestr	195	0.187 s	0.037 s	
ToolbarFactory>ToolbarFactory.getToolbar	1	0.172 s	0.121 s	
imageDisplayValidateParams	1	0.157 s	0.049 s	
isSingleImageDefaultPos	1	0.111 s	0.092 s	
num2str	780	0.099 s	0.064 s	
...nteractions.createDefaultInteractions	9	0.084 s	0.068 s	
...cribeLayer>ScribeLayer.ScribeLayer	3	0.082 s	0.043 s	
...alidateParams>getInterpolationType	1	0.075 s	0.074 s	
strcat	195	0.063 s	0.060 s	
...>PostUpdatePlugin.PostUpdatePlugin	1	0.053 s	0.053 s	
...:DesktopToolbarController.getInstance	1	0.051 s	0.000 s	

Figure 4.2: Profiling Results for the USTB Implementation of the Doppler Frequency Estimator

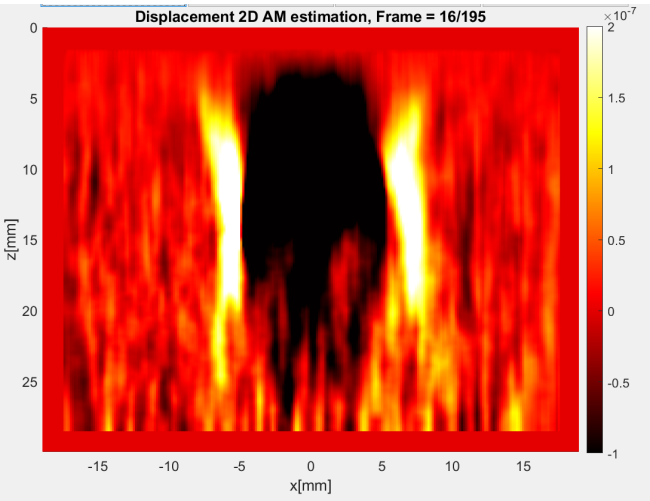
4.2.2. Performance of 2D AM

Figure 4.3 shows the displacement frames produced by applying the Two Dimensional Analytic Minimization (2D AM) displacement technique to the CIRS Spherical Phantom dataset. Similar to that of the Doppler estimator in Section 4.2.1, these displacement frames show that the 2D AM favours lower frequencies, which is inline with this technique's property of robustness with respect to large local decorrelations.

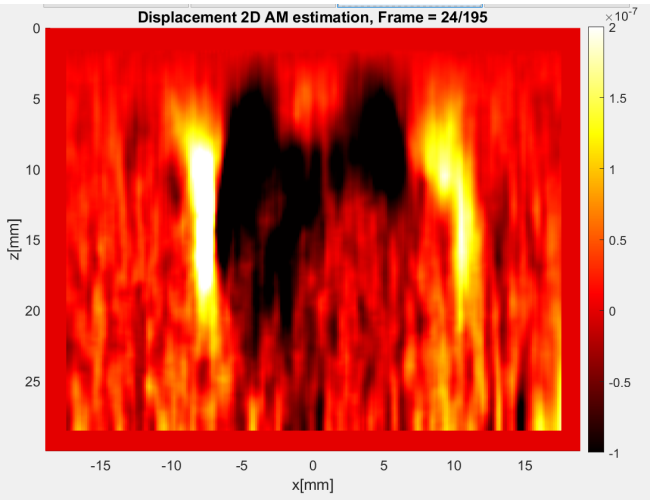
It is evident that the 2D AM results in smoother displacements than the Doppler estimator in Section 4.2.1. However this leads to less defined motion, as observed when visually tracking the wave-front travelling in Figure 4.1 and Figure 4.3. The ripple motion that is quite visible from Figure 4.1c to Figure 4.1d is more difficult to detect between Figure 4.3c and Figure 4.3d. Unlike the Doppler estimator, the wave motion in the 2D AM displacement frames are not clearly detectable until the 4^{th} or 5^{th} frame. This is partly due to this technique's need for more iterations of the cost function in Eq. 2.18 in Section 2.1.2 before the algorithm can adequately process the dataset in question. Figure 4.4 shows the MATLAB profiling results for the run time of the 2D AM, boxed in red. When applied to the CIRS Spherical Phantom dataset, the 2D AM takes 5.688 seconds, which is faster than Doppler frequency estimation covered in the previous section.



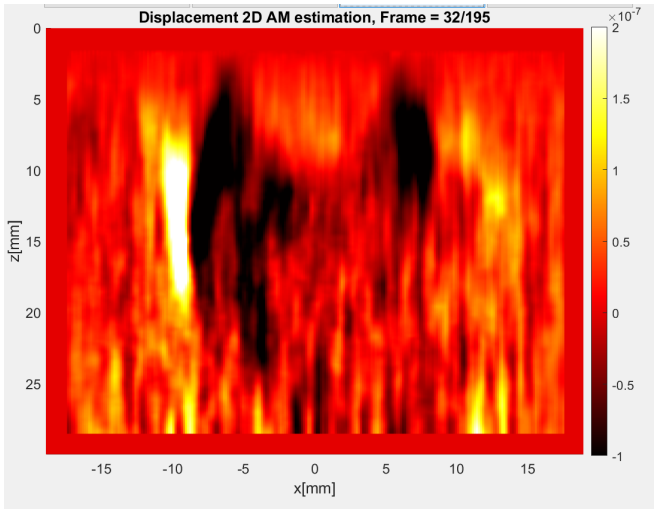
(a) Frame 8



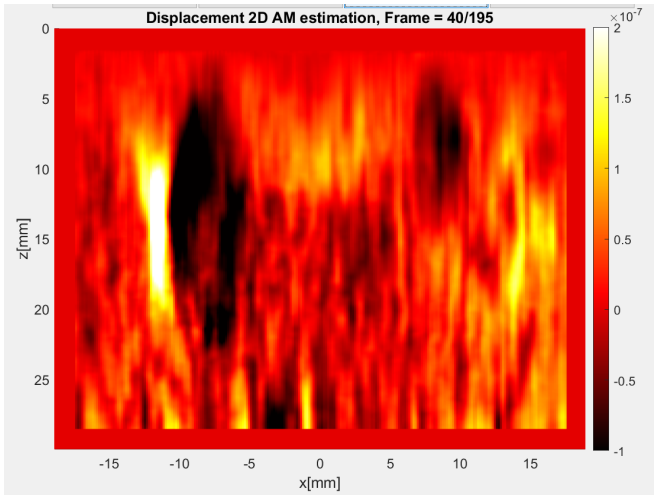
(b) Frame 16



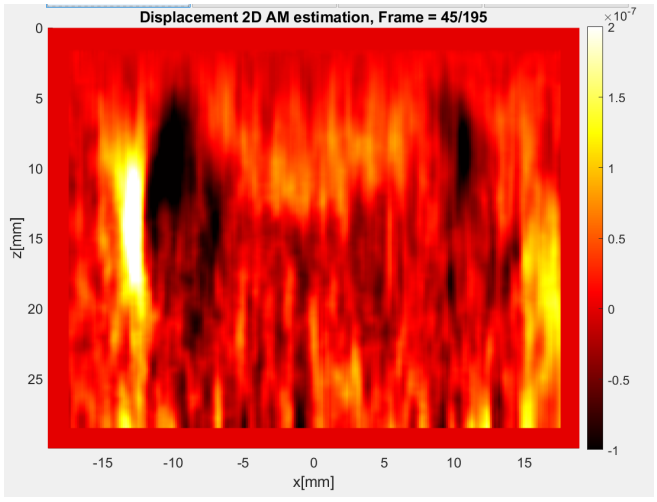
(c) Frame 24



(d) Frame 32



(e) Frame 40



(f) Frame 45

Figure 4.3: Displacement frames obtained using the 2D AM [19]

Profile Summary

Generated 06-Nov-2021 15:49:56 using performance time.

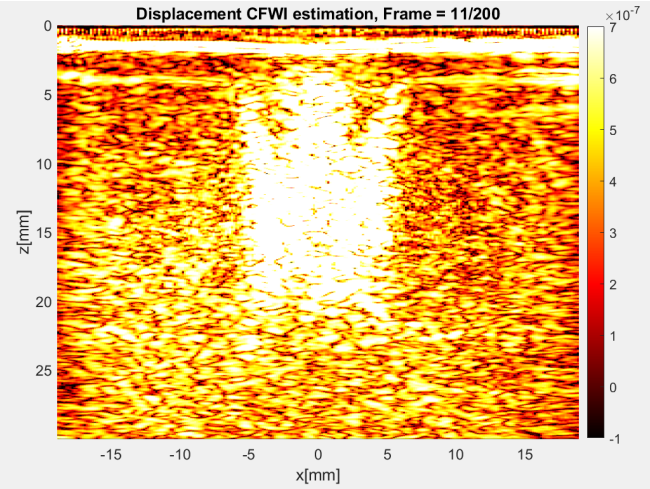
Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
my2dAMfunc	1	15.488 s	9.773 s	
AM2D (MEX-file)	180	5.688 s	5.688 s	
calc_att	1	0.026 s	0.026 s	
linear_scan>linear_scan.get.N_z_axis	1	0.000 s	0.000 s	
linear_scan>linear_scan.get.N_x_axis	1	0.000 s	0.000 s	

Figure 4.4: Profiling of the 2D AM Implementation Provided by the Authors of [19]

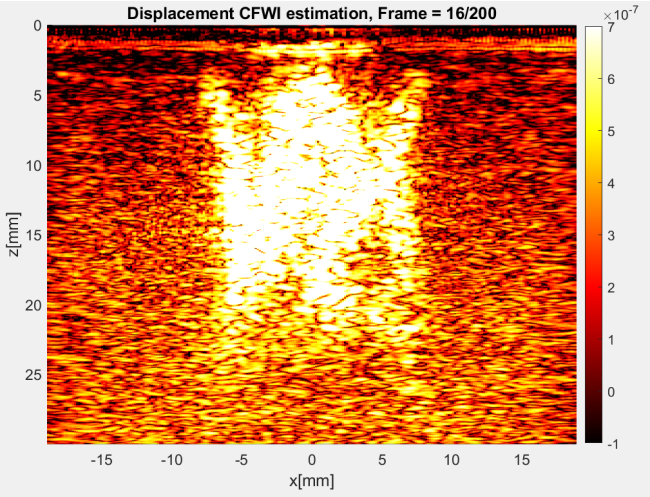
4.2.3. Performance of CFWI

Figure 4.5 shows the displacement frames of the Clutter Filter Wave Imaging (CFWI) displacement technique applied to the CIRS Spherical Phantom dataset. The CFWI aims to highlight relevant frequencies in a much more direct approach than the other two displacement techniques studied in this Chapter by using a band-pass filter.

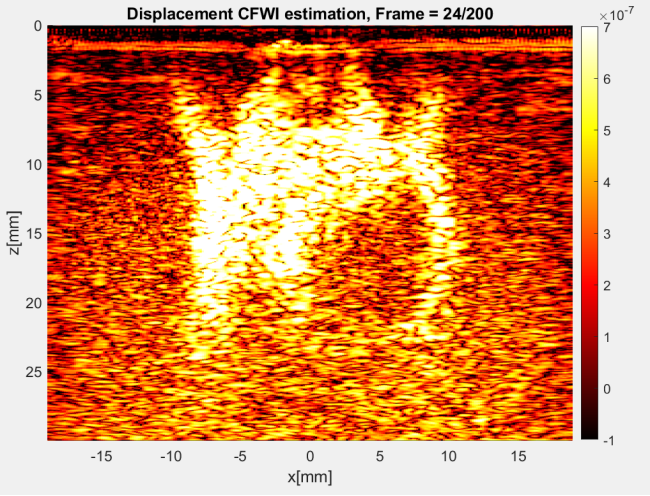
It is evident in Figure 4.5 that this approach is not as effective as the two previously discussed displacement techniques. High frequencies are still quite prominent, and the wavefront is difficult to see moving until the 11^{th} frame. The ripple motion from frame 24 to frame 32 in Figure 4.5c and Figure 4.5d, respectively, are more visible than that of the 2D AM in Figure 4.3, however it is obscured by the presence of high frequency noise, making the task of shear wave speed (SWS) estimators more difficult. Figure 4.6 shows that the run time of applying the CFWI to the CIRS Spherical Phantom dataset was 18.633 seconds.



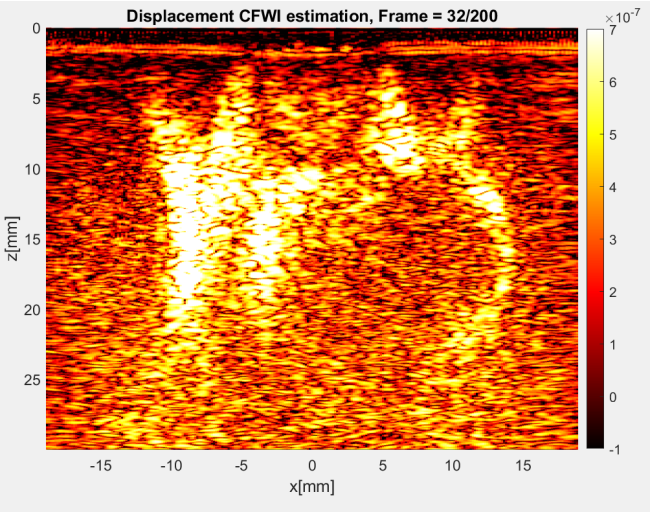
(a) Frame 11



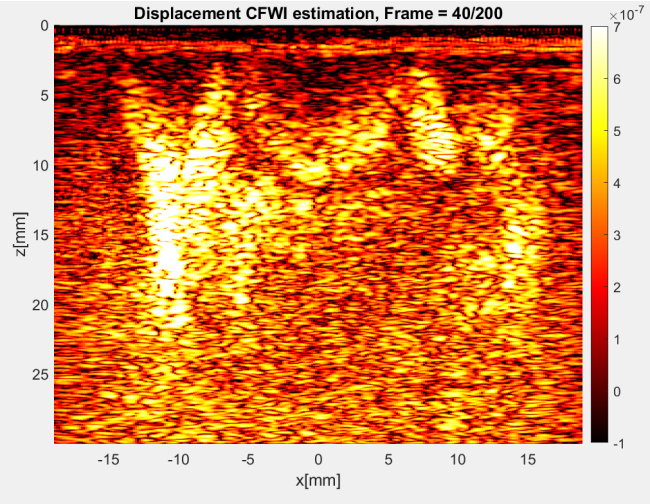
(b) Frame 16



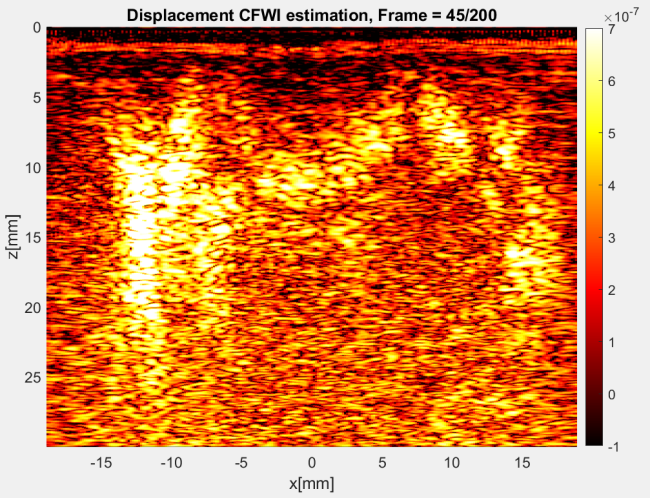
(c) Frame 24



(d) Frame 32



(e) Frame 40



(f) Frame 45

Figure 4.5: Displacement frames obtained using an implementation of the CFWI

Profile Summary

Generated 06-Nov-2021 15:52:10 using performance time.

Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
mycfwifunc	1	18.633 s	12.995 s	
butter	2	4.720 s	0.556 s	
lp2hp	1	3.585 s	3.577 s	
hilbert	180	0.918 s	0.024 s	
hilbert>hilbert_ml	180	0.892 s	0.880 s	
buttap	2	0.288 s	0.162 s	
zp2ss	2	0.127 s	0.083 s	
signal\private\isodd	2	0.106 s	0.067 s	
butter>buttnum	2	0.101 s	0.100 s	
bilinear	2	0.040 s	0.032 s	
isodd	2	0.038 s	0.022 s	
sigcasttofloat	6	0.030 s	0.018 s	
cplxpair	4	0.028 s	0.027 s	
poly	4	0.018 s	0.018 s	
sigcheckfloattype	22	0.017 s	0.017 s	
errorIf	2	0.017 s	0.017 s	
zp2ss>parse_input	2	0.015 s	0.015 s	
shiftdim	362	0.013 s	0.013 s	
lp2lp	1	0.003 s	0.001 s	
target	182	0.003 s	0.003 s	
signal\private\iirchk	2	0.003 s	0.003 s	
abcdchk	4	0.002 s	0.002 s	
assert	2	0.000 s	0.000 s	
nullcopy	2	0.000 s	0.000 s	
linear_scan>linear_scan.get.N_z_axis	1	0.000 s	0.000 s	
linear_scan>linear_scan.get.N_x_axis	1	0.000 s	0.000 s	

Figure 4.6: Profiling Results for the CFWI Implementation

Table 4.2 summarizes the run time measurements of the three displacement techniques

studied in this chapter. The 2D AM gives the shortest time measured, however this could be partly due to its implementation as a MATLAB executable (MEX) file. Attempts were made to port the Doppler estimation technique and the CFWI to MEX using the MATLAB's Coder tool, however this did not result in any significant improvement.

Table 4.2: Run time measurements of three displacement estimation techniques under consideration

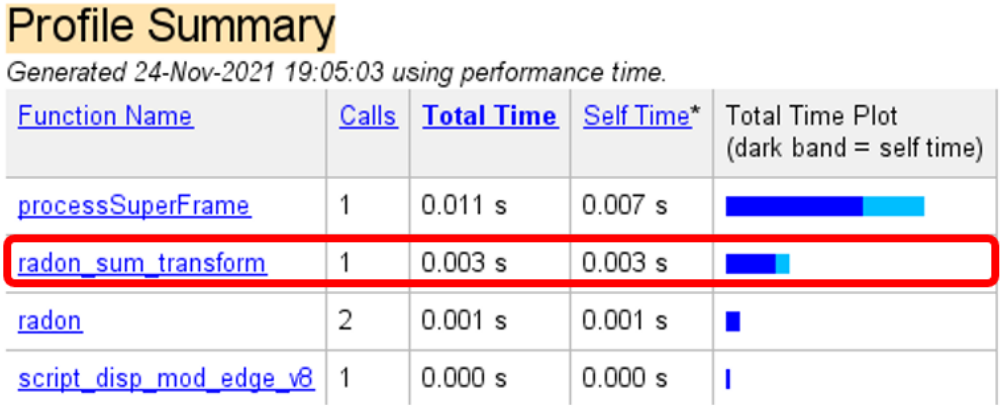
Displacement Estimation Technique	Run time (seconds)	Iterations	Time per iteration (seconds)
Doppler Frequency Estimation in the USTB	11.540	195	0.0592
2D AM	5.688	180	0.0316
CFWI	18.633	180	0.1035

4.3. Shear Wave Speed Estimates

The groups of displacement frames obtained from each of the techniques evaluated in Section 4.2 were used to generate SWS images based on the Linear Radon Transform (LRT), the

Radon Sum Transform (RST), and other SWS estimation techniques, such as linear regression [43], RANSAC [44], and FDSM [45] mentioned in Section 2.3. However, only the RST was able to adequately handle the noisy dataset under consideration and produce more or less acceptable SWS images. Therefore, this section is focused on the RST-based results, analysing the measured run time and estimated velocities.

The RST helps find trajectories that yield the largest sum of displacement values in a manner that is only restricted by positive change in frame or t axis. This property proved to be effective for the purpose of SWS estimation in the presence of large amounts of noise. Figure 4.7 shows that the run time of estimating a single velocity from a lateral window of 3 *mm* using the RST is approximately 0.003 seconds.



Profile Summary

Generated 24-Nov-2021 19:05:03 using performance time.

Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
processSuperFrame	1	0.011 s	0.007 s	
radon_sum_transform	1	0.003 s	0.003 s	
radon	2	0.001 s	0.001 s	
script_disp_mod_edge_v8	1	0.000 s	0.000 s	

Figure 4.7: Profiling Results for a Single Application of the Radon Sum Transform

The three experimental cases discussed in Table 4.1 of Section 4.1.4 were considered. The quality of the SWS image obtained in Case 1 from Table 4.1, which involved utilizing the CFWI displacement estimation technique, was quite far from the ideal SWS image, as illustrated in Figure 4.8. This is due to the large presence of high frequency noise from insufficient filtering by the clutter filter used in the CFWI.

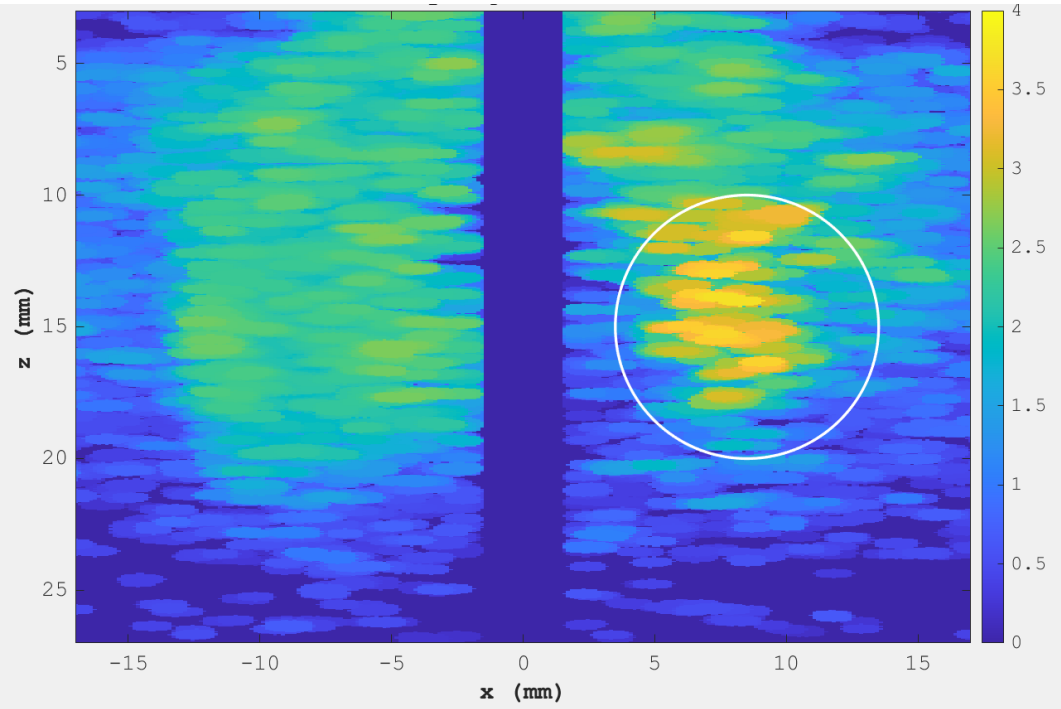


Figure 4.8: Radon Sum with Canny Edge Detection Mask on CFWI Frames

The color bar on the right-hand side maps the image to the obtained speed estimates in m/s. The white circle indicates the expected position of the lesion. An ideal SWS image for this dataset would contain velocities of 5.16 m/s and 2.89 m/s inside and outside of the white circle, respectively.

Figures 4.9 and 4.10 show the corresponding SWS images obtained in Cases 2 and 3, respectively. These images are closer to the ideal SWS image for the CIRS Spherical Phantom dataset. The ideal SWS image in question would have all velocities inside the white circle equal 5.16 m/s and all velocities outside the white circle equal 2.89 m/s. These values are obtained from the documented Young's modulus under the assumption of a homogeneous material density ρ of 1000 kg/m^3. The color intensity bar on the right sides of Figures 4.8, 4.9 and 4.10 are the magnitudes of the velocities in m/s.

In Case 2, the SWS estimator benefits from the regularization term in the 2D AM cost function: local correlation is mostly kept and there are very few drastic changes in neighbouring velocities. However, areas that contain zero or high velocities are the regions where the algorithm could not separate the Doppler frequency from noise, and Figure 4.9 appears

to have a large number of such regions.

In Case 3, the Doppler frequency displacement estimation algorithm based on auto-correlation was used instead. This resulted in less robustness towards local decorrelation, but Figure 4.10 shows that this case had more accuracy in calculating background velocities compared Case 2 in Figure 4.9.

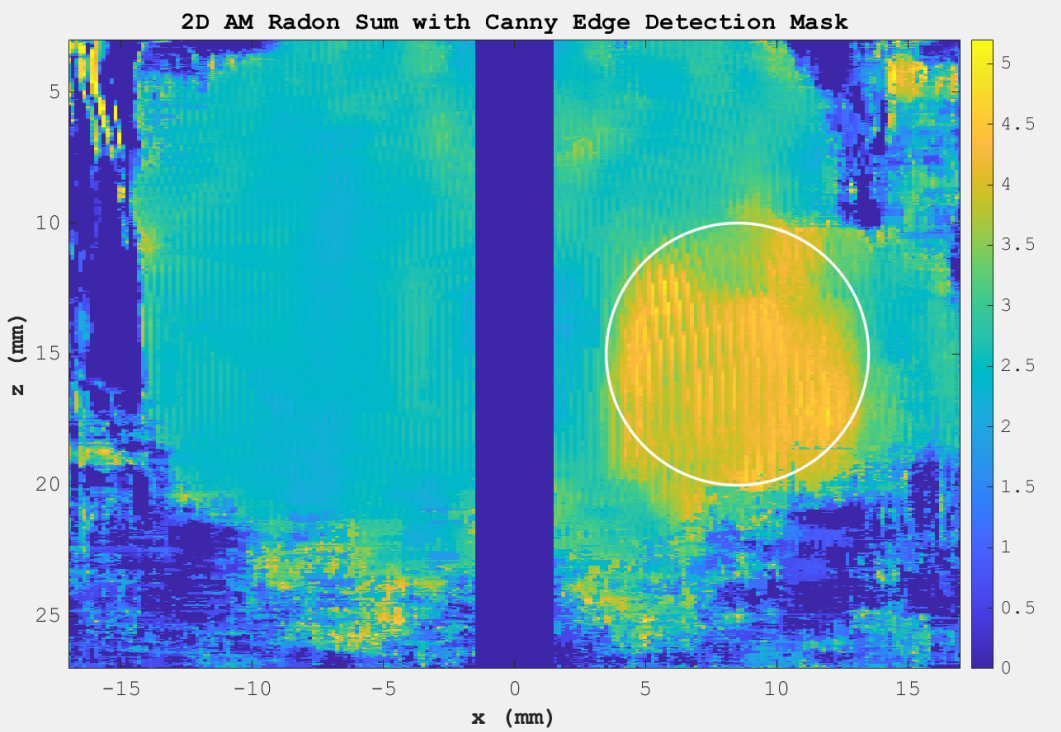


Figure 4.9: Radon Sum with Canny Edge Detection Mask on 2D AM Frames

The color bar on the right-hand side maps the image to the obtained speed estimates in m/s. The white circle indicates the expected position of the lesion. An ideal SWS image for this dataset would contain velocities of 5.16 m/s and 2.89 m/s inside and outside of the white circle, respectively.

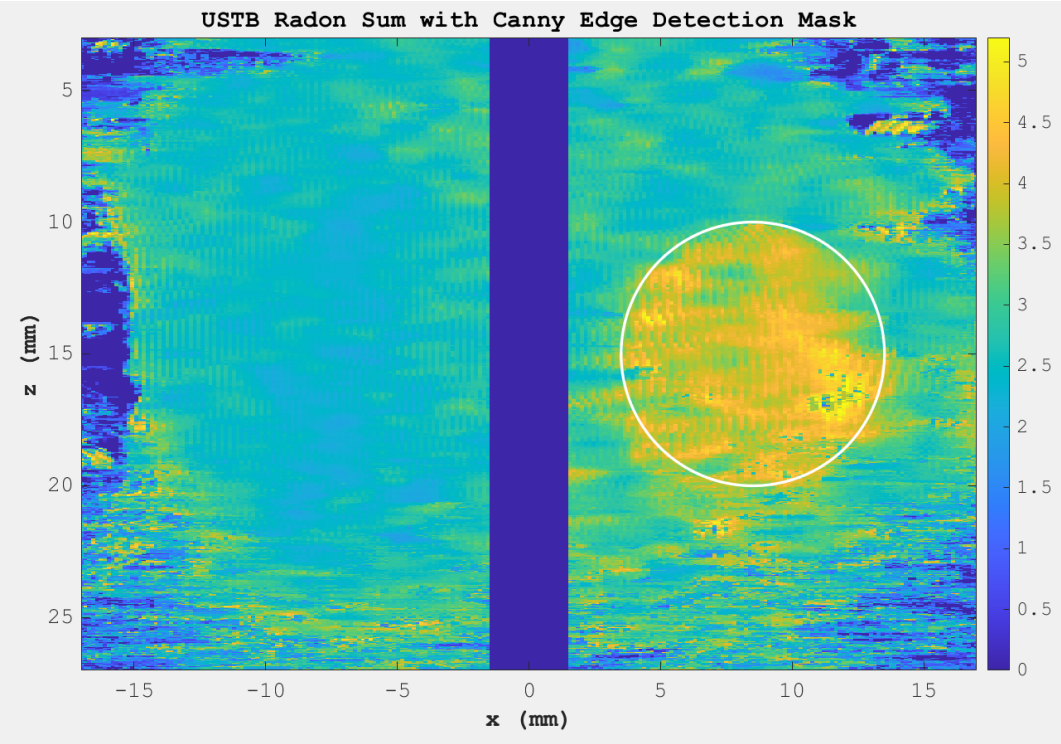


Figure 4.10: Radon Sum with Canny Edge Detection Mask on Doppler Estimation Frames

The color bar on the right-hand side maps the image to the obtained speed estimates in m/s. The white circle indicates the expected position of the lesion. An ideal SWS image for this dataset would contain velocities of 5.16 m/s and 2.89 m/s inside and outside of the white circle, respectively.

4.4. Morphological Filtering of SWS Estimates

The opening operation implemented in MATLAB's *imopen* function was applied to Case 2 (2D AM + Radon Sum) and Case 3 (Doppler frequency + Radon Sum) as indicated in Table 4.1 of Section 4.1.4. The *imopen* operator was used to achieve a blurring effect to promote 2D spatial continuity of SWS estimates. A binary disk served as the structuring element (SE) so that no particular spatial dimension was emphasized. Based on the typical observed size of pixel groups considered as noise, which are correlated to the width of each probing element of the Verasonics research scanner, the chosen radius of the SE for the *imopen* operator was set to 0.742 mm.

Figure 4.11 shows the runtime of applying MATLAB's *imopen* function to the entire $Z \times X$ mm^2 SWS image obtained in Section 4.3, where steps of Δz and Δx are 3.9113×10^{-2} mm and 1.4842×10^{-1} mm, respectively. Performing MATLAB's *imopen* morphological operation on $206,323$ SWS values takes approximately 6.580 seconds, or equivalently, 3.189×10^{-5} seconds per value. Inclusion of this filter entails relatively minor computational overhead, yet it improves image quality significantly as shown in Figures 4.12 and 4.13.

Profile Summary

Generated 24-Nov-2021 20:01:01 using performance time.

Function Name	Calls	**Total Time**	Self Time*	Total Time Plot (dark band = self time)
imopen	1	6.580 s	0.001 s	
images\private\morphop_fast	1	6.579 s	6.573 s	
strel>strel.strel	2	0.014 s	0.005 s	
strel>MakeDiskStrel	1	0.011 s	0.002 s	
strel>ParseInputs	1	0.007 s	0.004 s	
validatestring	1	0.003 s	0.001 s	
settings	1	0.002 s	0.001 s	
strel>strel.isflat	1	0.002 s	0.002 s	
settings	1	0.002 s	0.002 s	
validatestring>checkString	1	0.002 s	0.002 s	
target	1	0.001 s	0.001 s	
validatestring>checkInputs	1	0.001 s	0.001 s	
strel>strel.get.Neighborhood	2	0.001 s	0.001 s	
meshgrid	1	0.000 s	0.000 s	

Figure 4.11: Profiling Results for the Morphological Opening Operation

Figures 4.12 and 4.13 show the impact of applying MATLAB's *imopen* function to the results of the Radon Sum Transform with 2D AM displacement estimates (Figure 4.9) and

Doppler frequency based on auto-correlation displacement estimates (Figure 4.10), respectively. This operation aided in emphasizing the inclusion, filling continuity gaps, and allowing for a smoother transition from high to low SWS values and vice versa. Since this is a smoothing operation, the drawback is the reduction in the range of velocity values. Velocities that were ranging from 5.0 to 5.2 m/s in Figures 4.9 and 4.10 have been reduced to around 3.8 to 4.0 m/s in Figures 4.12 and 4.13. Consequently, such filtering should not be used for precise velocity estimation, but rather for highlighting the location and boundaries of inclusions in a cost effective manner.

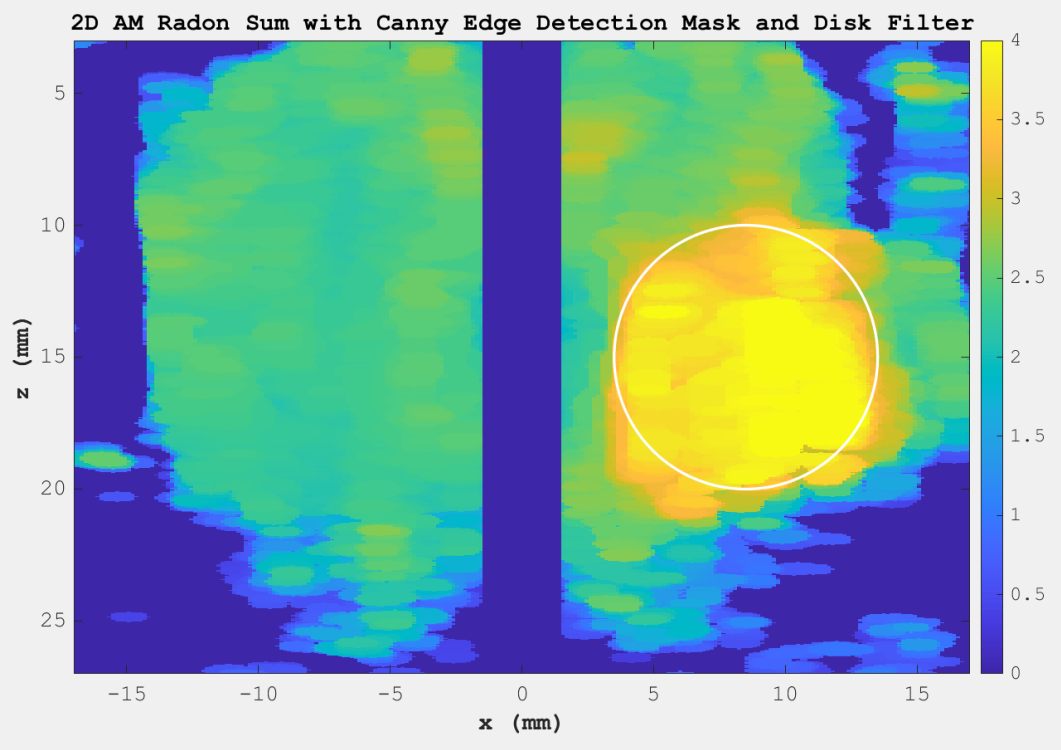


Figure 4.12: SWS Image after Morphological Filtering in Case 2

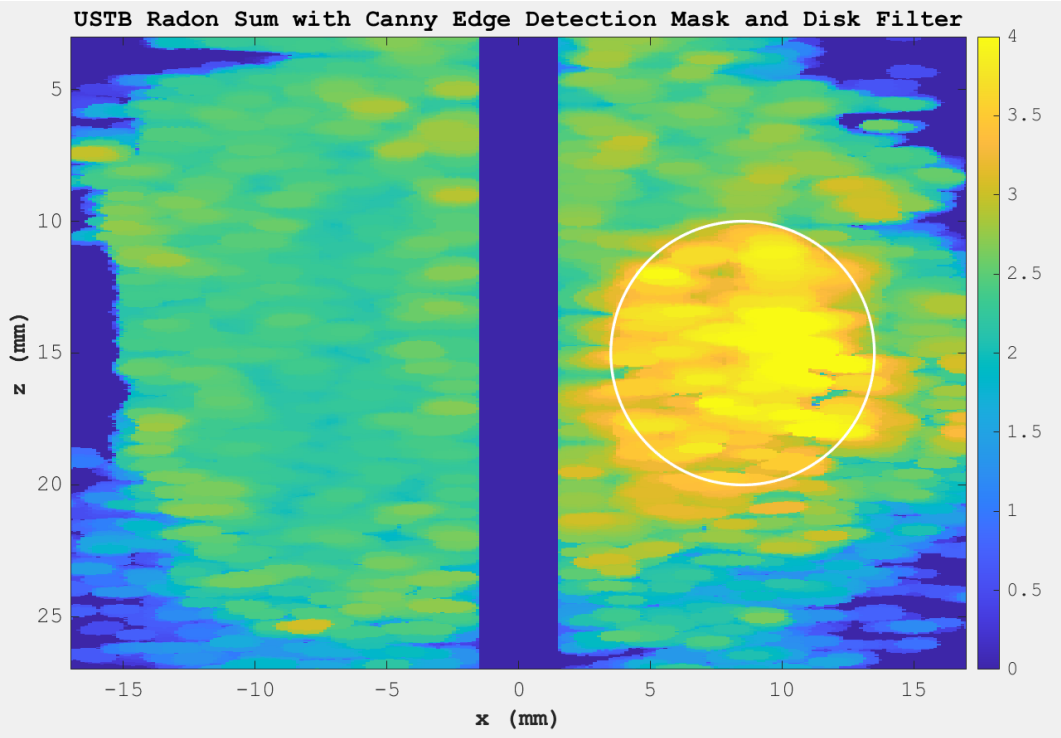


Figure 4.13: SWS Image after Morphological Filtering in Case 3

5. CONCLUDING REMARKS

Shear Wave Elastography, the process of extracting bio-mechanical properties such as the Young's modulus and Poisson's ratio in an imaged area of interest (AOI), was studied in this report. This modality relies on tracking the tissue movement caused by internal or external stimuli. Several sources of stimuli, such as heat, vibration shear waves, quasi-static compression, and dynamic compression have been explored by researchers and used for different applications. The report was focused on acoustic radiation force impulse (ARFI) excitation and subsequent tracking by means of ultrasound plane-wave imaging. The reported evaluation results are based on a public-domain dataset shared by the team behind the UltraSound ToolBox (USTB). The test medium under investigation was a CIRS Elasticity QA Spherical Phantom containing eight spheres immersed in a tissue-mimicking gel.

5.1. Summary

Experimentation with several processing stages in the Elastography pipeline was conducted. Early stages in the pipeline such as plane-wave data acquisition and delay-and-sum beamforming were outside the scope of the report. Instead, the project dealt with challenges associated with displacement estimation and subsequent shear wave speed estimation.

The Two-Dimensional Analytic Minimization (2D AM) technique approached displacement estimation by optimizing a cost function to first acquire integer displacement frames using a Dynamic Programming (DP) process, followed by solving another cost function with a regularization term to acquire axial and lateral sub-samples within those frames. The regularization term aided with robustness towards large local decorrelations, giving greater weight to similar patterns. The Doppler frequency estimator implemented as part of the USTB library utilized the phase of an auto-correlation function and the duration of a transmit event to

acquire the mean Doppler angular velocity. It was used to find the mean Doppler frequency shifts, which allowed us to calculate mean velocity values. The benefits of the two aforementioned correlation approaches were compared to the Clutter Filter Wave Imaging (CFWI) technique, which involved applying a simple temporal clutter filter to the beamformed IQ data. This provided clear visibility of the motion of interest by highlighting attenuated or low frequency bands while disregarding static data. However, high frequency noise was still quite pronounced in the displacement estimates.

The performance of the 2D AM and the Doppler frequency estimator were comparable, as both reduced high frequency noise effectively. Visually, the 2D AM produced smoother displacement estimates than the Doppler frequency estimator. However, the Doppler frequency estimator produced a more defined or sharper shear wavefront motion captured by displacement frames. MATLAB's code profiling tool indicated that the 2D AM was faster than the Doppler and CFWI estimators.

Displacement estimation was followed by shear wave speed (SWS) estimation using (x, t) data slices at individual depths z, where x is lateral displacement and t is time elapsed after ARFI excitation. This process relied on applying the sliding-window Radon Sum Transform after applying the Canny edge detection mask, both of which are novel aspects of this work. Other displacement and SWS estimation algorithms were explored, such as linear regression [43], Random Sample Consensus (RANSAC) [44], and Fourier-Domain Shift Matching (FDSM) [45]. However, the results obtained from applying these techniques were not satisfactory. In the same work that presented the 2D AM [19], the authors utilized a Kalman filter with a least squares linear regression speed estimator. Their publicated implementation of this speed estimator did not result in any noticeable improvements.

The performance of SWS estimation using CFWI, 2D AM, and auto-correlation displacement data (Cases 1, 2, and 3) was evaluated in Section 4.3. The run time was reported and the resulting SWS images were presented and compared to the ideal SWS image for the dataset

used. Section 4.4 described the impact of using the *imopen* morphological filtering operation implemented in MATLAB with a disk structuring element. The filter helped emphasize the appearance of the lesion, and filled spatial continuity gaps. The run time of applying this morphological filter was reported as well. The drawback was that the filter reduced the range of the speed estimates.

5.2. Future Work

One of the latest works on SWS estimation developed a method based on solving the Eikonal equation to achieve high accuracy in the presence of noise [53]. This method, referred to as the tangent plane method (TPM) was applied to the same CIRS Elasticity QA Spherical Phantom used in this project. The authors compared its performance to a two-dimensional cross correlation method and a distance method. The TPM method involved some post-processing of SWS images as well [53], using a spatial Gaussian filter and a temporal FIR low-pass filter. Similar to this project, they also took advantage of the benefits of using a directional filter. Future efforts could be made to investigate how the use of the Canny edge detection mask strategy presented in this project could improve the performance of the TPM method.

Further work could involve experimenting with more complex structuring elements (SE) rather than a simple disk. The work in [56] discusses the use of a dynamic SE in order to preserve objects intersecting unwanted line segments regardless of orientation. Another natural next step would be to investigate diagnostic potential of SWS images and elastograms. Numerous works have trained machine learning (ML) models on outputs of different stages of the Elastography pipeline [57]–[59]. Their results show methods that utilize neural networks and other ML schemes outperform traditional diagnostic approaches. Efforts to extend this project could be made to further investigate and incorporate one of these ML techniques to extracting tissue health information from SWS images.

BIBLIOGRAPHY

[1] K. J. Parker, L. S. Taylor, S. Gracewski, and D. J. Rubens, "A unified view of imaging the elastic properties of tissue," *The Journal of the Acoustical Society of America*, vol. 117, no. 5, pp. 2705–2712, 2005.

[2] G. Montaldo, M. Tanter, J. Bercoff, N. Benech, and M. Fink, "Coherent plane-wave compounding for very high frame rate ultrasonography and transient elastography," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 56, no. 3, pp. 489–506, 2009.

[3] J. Ophir, S. Srinivasan, R. Righetti, and A. Thittai, "Elastography: A decade of progress (2000-2010)," *Current Medical Imaging*, vol. 7, no. 4, pp. 292–312, 2011.

[4] M. Mirzaei, A. Asif, and H. Rivaz, "Accurate and precise time-delay estimation for ultrasound elastography with prebeamformed channel data," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 67, no. 9, pp. 1752–1763, 2020.

[5] J. Bercoff, M. Tanter, and M. Fink, "Supersonic shear imaging: A new technique for soft tissue elasticity mapping," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 51, no. 4, pp. 396–409, 2004.

[6] Y. H. Yoon, S. Khan, J. Huh, and J. C. Ye, "Efficient B-mode ultrasound image reconstruction from sub-sampled RF data using deep learning," *IEEE Transactions on Medical Imaging*, vol. 38, no. 2, pp. 325–336, 2018.

[7] R. J. Van Sloun, R. Cohen, and Y. C. Eldar, "Deep learning in ultrasound imaging," *Proceedings of the IEEE*, vol. 108, no. 1, pp. 11–29, 2019.

[8] M. Albulayli and D. Rakhmatov, "Fourier domain depth migration for plane-wave ultrasound imaging," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 65, no. 8, pp. 1321–1333, 2018.

[9] E. Macé, G. Montaldo, I. Cohen, M. Baulac, M. Fink, and M. Tanter, "Functional ultrasound imaging of the brain," *Nature Methods*, vol. 8, no. 8, pp. 662–664, 2011.

[10] M. Tanter and M. Fink, "Ultrafast imaging in biomedical ultrasound," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 61, no. 1, pp. 102–119, 2014.

[11] A. Besson *et al.*, "A sparse reconstruction framework for Fourier-based plane-wave imaging," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 63, no. 12, pp. 2092–2106, 2016.

[12] J. J. Mai and M. F. Insana, "Strain imaging of internal deformation," *Ultrasound in Medicine & Biology*, vol. 28, no. 11-12, pp. 1475–1484, 2002.

[13] K. J. Parker, "The evolution of vibration sonoelastography," *Current Medical Imaging Reviews*, vol. 7, no. 4, pp. 283–291, 2011.

[14] W. L. Donegan, "Evaluation of a palpable breast mass," *New England Journal of Medicine*, vol. 327, no. 13, pp. 937–942, 1992.

[15] R. J. van Sloun, R. R. Wildeboer, H. Wijkstra, and M. Mischi, "Viscoelasticity mapping by identification of local shear wave dynamics," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 64, no. 11, pp. 1666–1673, 2017.

[16] A. K. Tehrani and H. Rivaz, "Displacement estimation in ultrasound elastography using pyramidal convolutional neural network," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 67, no. 12, pp. 2629–2639, 2020.

[17] H. Rivaz, E. Boctor, P. Foroughi, R. Zellars, G. Fichtinger, and G. Hager, "Ultrasound elastography: A dynamic programming approach," *IEEE Transactions on Medical Imaging*, vol. 27, no. 10, pp. 1373–1377, 2008.

[18] H. S. Hashemi and H. Rivaz, "Global time-delay estimation in ultrasound elastography," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 64, no. 10, pp. 1625–1636, 2017.

[19] H. Rivaz, E. M. Boctor, M. A. Choti, and G. D. Hager, "Real-time regularized ultrasound elastography," *IEEE Transactions on Medical Imaging*, vol. 30, no. 4, pp. 928–945, 2010.

[20] O. M. H. Rindal, *Arfi from uff file recorded with verasonics*, Aug. 2017. [Online]. Available: http://www.ustb.no/examples/acoustical-radiation-force-imaging/arfi-from-uff-file-recorded-with-verasonics/.

[21] A. Rodriguez-Molares *et al.*, "The ultrasound toolbox," in *2017 IEEE International Ultrasonics Symposium (IUS)*, IEEE, 2017, pp. 1–4.

[22] C. Kasai, K. Namekawa, A. Koyano, and R. Omoto, "Real-time two-dimensional blood flow imaging using an autocorrelation technique," *IEEE Transactions on Sonics and Ultrasonics*, vol. 32, no. 3, pp. 458–464, 1985.

[23] W. D. Barber, J. W. Eberhard, and S. G. Karr, "A new time domain technique for velocity measurements using doppler ultrasound," *IEEE Transactions on Biomedical Engineering*, vol. BME-32, no. 3, pp. 213–229, 1985.

[24] B. A. Angelsen and K. Kristoffersen, "Discrete time estimation of the mean Doppler frequency in ultrasonic blood velocity measurements," *IEEE Transactions on Biomedical Engineering*, vol. BME-30, no. 4, pp. 207–214, 1983.

[25] D. A. Christopher, P. N. Burns, J. Armstrong, and F. S. Foster, "A high-frequency continuous-wave Doppler ultrasound system for the detection of blood flow in the microcirculation," *Ultrasound in Medicine & Biology*, vol. 22, no. 9, pp. 1191–1203, 1996.

[26] C. Hartley and J. Cole, "An ultrasonic pulsed Doppler system for measuring blood flow in small vessels.," *Journal of Applied Physiology*, vol. 37, no. 4, pp. 626–629, 1974.

[27] D. L. Franklin, W. Schlegel, and R. F. Rushmer, "Blood flow measured by Doppler frequency shift of back-scattered ultrasound," *Science*, vol. 134, no. 3478, pp. 564–565, 1961.

[28] C. F. Hottinger and J. D. Meindl, "Blood flow measurement using the attenuation-compensated volume flowmeter," *Ultrasonic Imaging*, vol. 1, no. 1, pp. 1–15, 1979.

[29] D. W. Baker, "Pulsed ultrasound doppler bound flow sensing," *IEEE Transactions on Sonics and Ultrasonics*, vol. SU-17, pp. 170–185, 1970.

[30] P. P. Lee and R. C. Waag, "Spectral analysis of pulsed ultrasound for blood flow measurement in the presence of noise," in *1978 International Ultrasonics Symposium*, IEEE, 1978, pp. 353–356.

[31] M. Arts and J. Roevros, "On the instantaneous measurement of bloodflow by ultrasonic means," *Medical and Biological Engineering*, vol. 10, no. 1, pp. 23–34, 1972.

[32] R. Pawula, "Analysis of an estimator of the center frequency of a power spectrum," *IEEE Transactions on Information Theory*, vol. 14, no. 5, pp. 669–676, 1968.

[33] B. A. Angelsen and A. Brubakk, "Transcutaneous measurement of blood flow velocity in the human aorta," *Cardiovascular Research*, vol. 10, no. 3, pp. 368–379, 1976.

[34] B. A. Angelsen, "Instantaneous frequency, mean frequency, and variance of mean frequency estimators for ultrasonic blood velocity Doppler signals," *IEEE Transactions on Biomedical Engineering*, vol. BME-28, no. 11, pp. 733–741, 1981.

[35] W. R. Brody and J. D. Meindl, "Theoretical analysis of the CW Doppler ultrasonic flowmeter," *IEEE Transactions on Biomedical Engineering*, vol. BME-21, no. 3, pp. 183–192, 1974.

[36] T. Loupas, J. Powers, and R. W. Gill, "An axial velocity estimator for ultrasound blood flow imaging, based on a full evaluation of the Doppler equation by means of a two-dimensional autocorrelation approach," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 42, no. 4, pp. 672–688, 1995.

[37] S. Salles, L. Løvstakken, S. A. Aase, T. G. Bjåstad, and H. Torp, "Clutter filter wave imaging," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 66, no. 9, pp. 1444–1452, 2019.

[38] Y. Luo, J. Xia, R. D. Miller, Y. Xu, J. Liu, and Q. Liu, "Rayleigh-wave dispersive energy imaging using a high-resolution linear Radon transform," *Pure and Applied Geophysics*, vol. 165, no. 5, pp. 903–922, 2008.

[39] N. C. Rouze, M. H. Wang, M. L. Palmeri, and K. R. Nightingale, "Robust estimation of time-of-flight shear wave speed using a Radon sum transformation," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 57, no. 12, pp. 2662–2670, 2010.

[40] O. M. H. Rindal, *Datasets - ultrasound toolbox. acoustical radiation force imaging*, Aug. 2017. [Online]. Available: http://www.ustb.no/ustb-datasets/.

[41] CIRS, *CIRS QA spherical phantom data sheet*, 2016. [Online]. Available: http://www.cirsinc.com/wp-content/uploads/2019/04/049-049-DS-120418.pdf.

[42] A. Besson, R. E. Carrillo, O. Bernard, Y. Wiaux, and J.-P. Thiran, "Compressed delay-and-sum beamforming for ultrafast ultrasound imaging," in *2016 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2016, pp. 2509–2513.

[43] A. Evans *et al.*, "Can shear-wave elastography predict response to neoadjuvant chemotherapy in women with invasive breast cancer?" *British Journal of Cancer*, vol. 109, no. 11, pp. 2798–2802, 2013.

[44] M. H. Wang, M. L. Palmeri, V. M. Rotemberg, N. C. Rouze, and K. R. Nightingale, "Improving the robustness of time-of-flight based shear wave speed reconstruction methods using RANSAC in human liver in vivo," *Ultrasound in Medicine & Biology*, vol. 36, no. 5, pp. 802–813, 2010.

[45] D. Rosen and J. Jiang, "Fourier-domain shift matching: A robust time-of-flight approach for shear wave speed estimation," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 65, no. 5, pp. 729–740, 2018.

[46] S. L. Lipman, N. C. Rouze, M. L. Palmeri, and K. R. Nightingale, "Evaluating the improvement in shear wave speed image quality using multidimensional directional filters in the presence of reflection artifacts," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 63, no. 8, pp. 1049–1063, 2016.

[47] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[48] T. Deffieux, J.-L. Gennisson, J. Bercoff, and M. Tanter, "On the effects of reflected waves in transient shear wave elastography," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 58, no. 10, pp. 2032–2035, 2011.

[49] P. Song, H. Zhao, A. Manduca, M. W. Urban, J. F. Greenleaf, and S. Chen, "Comb-push ultrasound shear elastography (CUSE): A novel method for two-dimensional shear elasticity imaging of soft tissues," *IEEE Transactions on Medical Imaging*, vol. 31, no. 9, pp. 1821–1832, 2012.

[50] P. Song, M. W. Urban, A. Manduca, H. Zhao, J. F. Greenleaf, and S. Chen, "Comb-push ultrasound shear elastography (CUSE) with various ultrasound push beams," *IEEE Transactions on Medical Imaging*, vol. 32, no. 8, pp. 1435–1447, 2013.

[51] E. Weisstein, *Radon transform*. [Online]. Available: https://mathworld.wolfram.com/RadonTransform.html.

[52] M. Ito, M. Tsubai, and A. Nomura, "Morphological operations by locally variable structuring elements and their applications to region extraction in ultrasound images," *Systems and Computers in Japan*, vol. 34, no. 3, pp. 33–43, 2003.

[53] H.-k. Lee, D. Kong, K. Choi, R. Mislati, and M. M. Doyley, "A robust and fast method for 2d shear wave speed calculation," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 68, no. 7, 2021.

[54] M. Tanter *et al.*, "Quantitative assessment of breast lesion viscoelasticity: Initial clinical results using supersonic shear imaging," *Ultrasound in Medicine & Biology*, vol. 34, no. 9, pp. 1373–1386, 2008.

[55] N. Gulley, "Picking up the pace with the matlab profiler," *MATLAB News & Notes*, 2003.

[56] J. N. Said, M. Cheriet, and C. Y. Suen, "Dynamical morphological processing: A fast method for base line extraction," in *13th International Conference on Pattern Recognition*, IEEE, vol. 2, 1996, pp. 8–12.

[57] Y. Chen *et al.*, "Machine-learning-based classification of real-time tissue elastography for hepatic fibrosis in patients with chronic hepatitis b," *Computers in Biology and Medicine*, vol. 89, pp. 18–23, 2017.

[58] Q. Zhang *et al.*, "Deep learning based classification of breast tumors with shear-wave elastography," *Ultrasonics*, vol. 72, pp. 150–157, 2016.

[59] M. Adel, A. Kotb, O. Farag, M. S. Darweesh, and H. Mostafa, "Breast cancer diagnosis using image processing and machine learning for elastography images," in *8th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, IEEE, 2019, pp. 1–4.