Nicole Haderer (1005721444), Deniz Uzun (1006035005)

ULTRASOUND IMAGING LAB

PART 1: IMAGE PROCESSING AND DISPLAY

21. Calculate the time recorded in each line and calculate the depth reached in each line.

The time recorded in each line is determined by dividing the number of data points taken per line (in the RawRF matrix) by the sampling frequency at which each data point is collected as shown below. These values correspond to "NumberOfDataPointsInFastTime" of 10000 data points/line (the horizontal dimension of the RawRF matrix) and "SampleFreq" of 500 MHz respectively.

$$\begin{aligned}\text{Time recorded in each line} &= \text{NumberOfDataPointsInFastTime}/\text{SampleFreq}\\ &= (10000 \text{ data points per line})/(500 \text{ MHz})\\ &= \mathbf{20\ \mu sec/line}\end{aligned}$$

The depth reached by each line can then be found by multiplying the time recorded per line by the speed of sound. Since the sound wave must travel the distance twice (to and from the reflected interface), this value must then be divided by 2. The speed of sound, "c", is assumed to be 1540 m/s.

$$\begin{aligned}\text{Depth reached in each line} &= (\text{Time recorded in each line})\times c/2\\ &= (20\ \mu\text{sec/line})\times(1540\ \text{m/s})/2\\ &= \mathbf{15.4\ mm/line}\end{aligned}$$

Therefore, it takes 20 µsec to collect all data points in each line with each reaching a depth of 15.4 mm. This depth will correspond to the vertical size of the image the data produces. Both calculations were performed in MATLAB as shown in Figure 1.

```
%% Question 21. Calculate the time recorded in each line and calculate the depth reached in each line
time_recorded = NumberOfDataPointsInFastTime / SampleFreq * 10^6; %in microseconds
depth_reached = time_recorded * c / 2 * 10^-3; %in mm
```

Figure 1. The MATLAB code used to calculate the time recorded and the depth reached by each line in the RawRF data matrix.

22. Plot a single A-line (as a function of time) using line 200. Produce an appropriate x axis (time). Also plot a zoom-in to the signal from 2 to 4 microseconds. Give your axes appropriate labels.

To plot the signal intensity data points against time, a time vector with 10000 (number of data points in a line) evenly spaced values ranging from 0 to 20 µsec (total time to record each line) was created using the linspace() function as shown in Figure 4. The 200th line of the RawRF matrix was then indexed and plotted against this time vector and the xlim() function was then used to view data in the range of 2 to 4 µsec.

```
%% Question 22. Plot a single A-line (as a function of time): use line 200
% produce an appropriate x axis (time)
% also plot a zoom-in to the signal from 2 to 4 microseconds
% give your axes appropriate labels

time_vector = linspace(0,time_recorded, NumberOfDataPointsInFastTime);
plot(time_vector, RawRF(:,200))
title('Single A-line (Line 200) as a Function of Time')
xlabel('Time (microseconds)')
ylabel('Signal Intensity')

%Zoom in
xlim([2 4])
```

Figure 2. The MATLAB code used to generate plots in Figures 3 & 4.

The measured signal intensities of line 200 of the RawRF data matrix are plotted in Figure 2 against the time taken to collect all data points in a line (20 µsec).

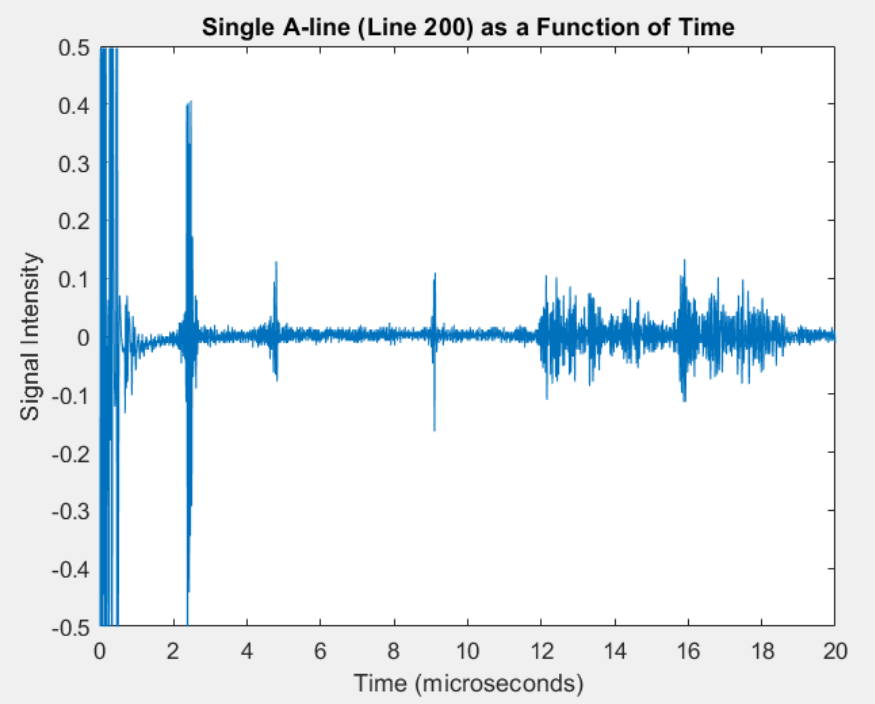

Figure 3. Signal intensity vs. time (in microseconds) plot of a single A-line (line 200 of RawRF).

A zoom-in of the signal intensities of line 200 from 2 to 4 µsec are plotted in Figure 4 below.

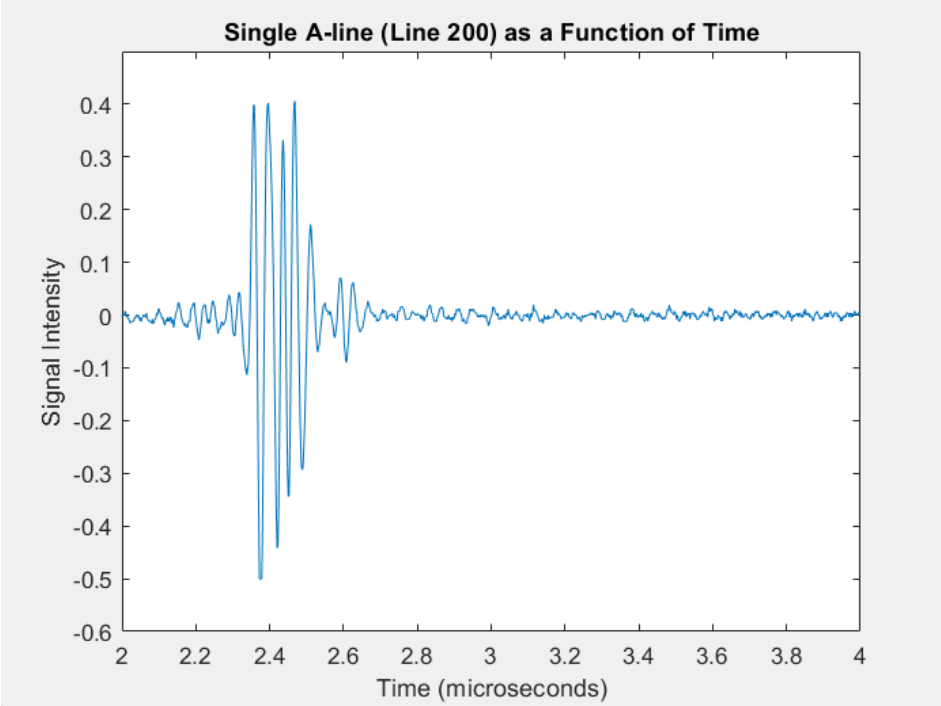

Figure 4. The signal intensity vs. time plot of a single A-line (line 200 of RawRF) in the 2 to 4 µsec time interval.

As can be seen in Figure 4, the greatest signal received is a sound wave of an amplitude of about 0.4, recorded at around 2.4 µsec during the collection of data in line 200.

23. Plot a raw B-mode image (in grayscale). Use the 'imagesc' function; use axes in mm. Use labels and a title. Indicate the position of the A-line plotted in item 22.

The data from the RawRF matrix was plotted in a raw B-mode image by first creating depth and lateral vectors representing the image dimensions. The linspace() function was used to create these vectors with evenly spaced values using the known image dimensions, 20 mm (lateral; represented by "LateralSize") by 15.4 mm (vertical; depth reached by each line), as well the RawRF data matrix dimensions, 10000 (vertical dimension of image; represented by "NumberOfDataPointsInFastTime") by 382 (lateral dimension of image; represented by the "NumberOfLines"). The RawRF matrix was then plotted against these vectors using the imagesc() function in grayscale as shown in Figure 6.

```
%% Question 23. Plot a raw B-mode image (in grayscale)
% use the 'imagesc' function; use axes in mm
% use labels and title
% indicate the position of the A-line plotted in item 22

depth_axis = linspace(0, depth_reached, NumberOfDataPointsInFastTime);
lateral_axis = linspace(0, LateralSize, NumberOfLines);

figure;
imagesc(lateral_axis, depth_axis, abs(RawRF));
colormap('gray'); % Set the colormap to grayscale
title('Raw B-Mode Image');
xlabel('Lateral Position (mm)');
ylabel('Depth (mm)');

hold on;
line([lateral_axis(200) lateral_axis(200)], [min(depth_axis) max(depth_axis)], 'Color', 'r');
hold off;
```

Figure 5. The MATLAB code used to generate the raw B-mode image in Figure 6.

The raw B-mode image is shown in Figure 6 below. The A-line (line 200 of RawRF) is indicated by a red line positioned just after 10 mm in the lateral axis.

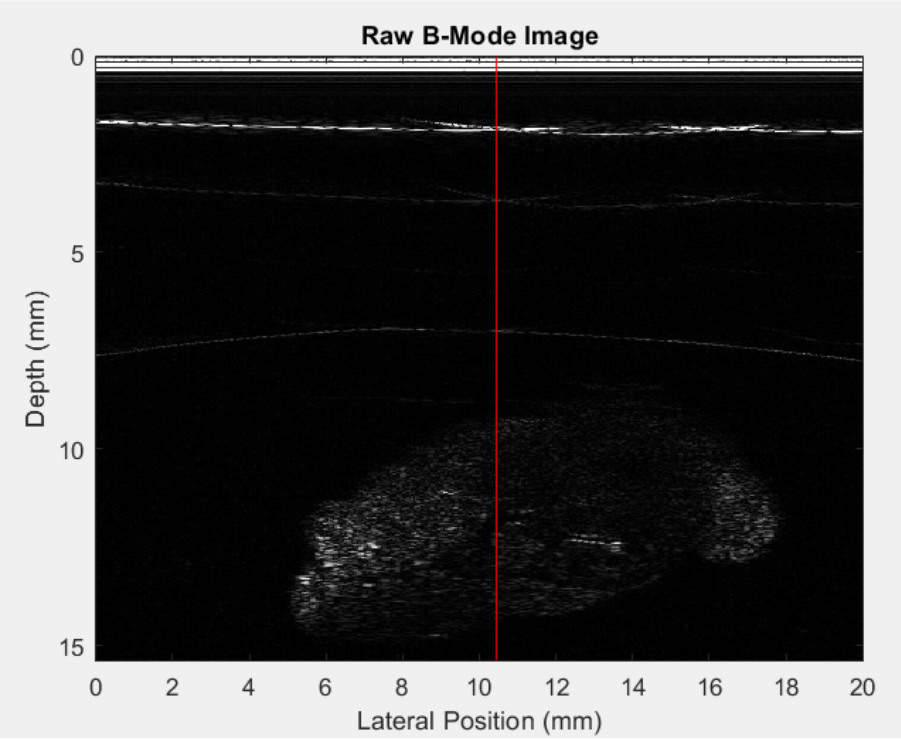

Figure 6. Raw B-mode image in grayscale. The red line indicates the position of the A-line (Line 200) plotted in Question 22.

24. Perform envelope detection. Envelope detection [one of the below two approaches suggested - a) use hilbert transform or b) use demodulation]. Describe qualitatively what changes in the image when you perform this processing step.

The Hilbert transform was used to perform envelope detection of the signals from each line of the RawRF matrix as shown in Figure 7. In this process, the overall amplitude in the received signal (shown in Figure 3 or 4 for example) is determined by enveloping the signal fluctuations. The Hilbert transform specifically does this by performing a 90˚ phase shift in the input signal such that the peaks are shifted half-way towards the troughs. The combination of this transformed signal with the original signal estimates the magnitude of the envelope [1].

```
%% Question 24. Perform envelope detection
% envelope detection [one of the below two approaches suggested - a) use hilbert transform or b) use demodulation]
% a) Hilbert transform (easier? refer to class notes)
%  use 'hilbert' function
% b) Demodulation approach:
% demodulate into in-phase and quadrature components and
% then calculate the envelope as sqrt(I^2+Q^2).
% Use the Matlab function 'demod' with the option 'qam' (need Signal
% Processing Toolbox)
% then plot the same (envelope-detected) A-line as before into the figures of step 2
% and plot a new B-mode image
%
% Describe qualitatively what changes in the image when you perform this
% processing step

figure;
imagesc(lateral_axis, depth_axis, abs(hilbert(RawRF)));
colormap('gray'); % Set the colormap to grayscale
title('B-Mode Image with Hilbert Transform');
xlabel('Lateral Position (mm)');
ylabel('Depth (mm)');
```

Figure 7. The MATLAB code used to generate the B-mode image with Hilbert Transform envelope detection in Figure 8.

The transformed matrix was plotted in the same way as in Question 23 and is shown in Figure 8.

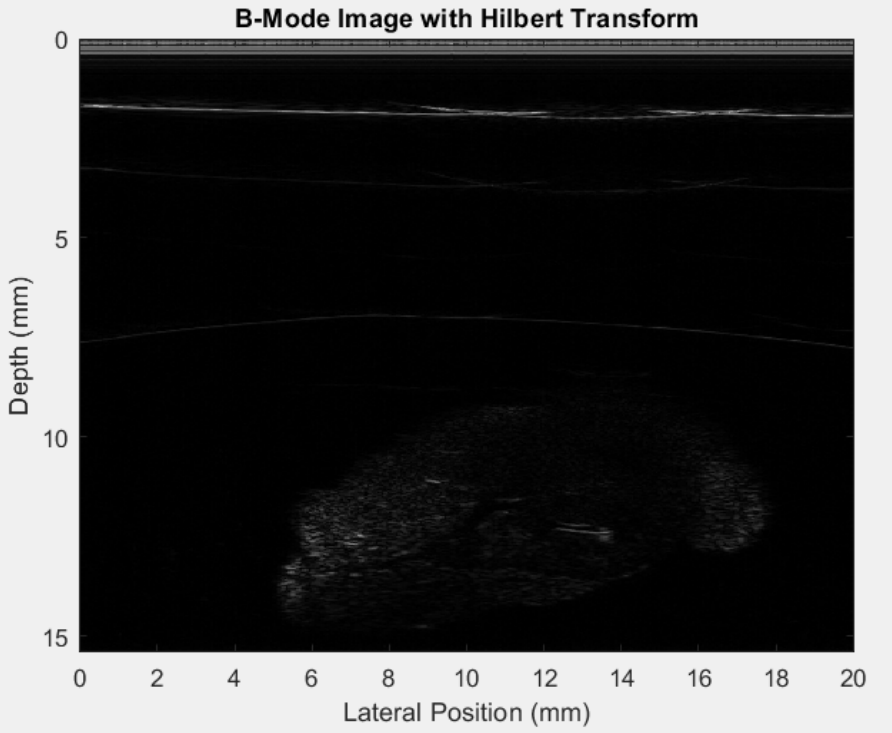

Figure 8. B-mode image after Hilbert Transform for envelope detection is performed.

In comparison with the raw B-mode image (Figure 6), the image with envelope detection has lower contrast between tissue structures and the background. Enveloping multiple peaks into one large peak causes this decrease in contrast. However, the rippling artifacts at the top and centre of the original image as well as "speckling" artifacts are significantly reduced, improving the image in terms of noise content. The reduction in speckling can be attributed to the "smoothening" of the baseline signal received by the transducer during enveloping; any small peaks close to the midline of the received signal (along the horizontal axis) are smoothed.

25. Perform 'log compression'. Take logarithm of data so that a large range of signal intensities can be visualised in one plot, and plot the result. Describe qualitatively what changes in the image when you perform this processing step. Identify the following features in the resulting image: kidney, near-field diffraction artifact, multiple reflection artifact.

Logarithmic compression is performed to amplify lower-amplitude signals in relation to higher-amplitude signals, making them appear brighter in the B-mode image. To logarithmically compress the image's signals, the log10() function was applied to the absolute values (using the abs() function) of the RawRF matrix as shown in Figure 9. The abs() function was applied before compression since logarithmic functions are undefined for inputs of 0 or less.

```
%% Question 25. Perform 'log compression'
% take logarithm of data so that a large range
% of signal intensities can be visualized in one plot, and plot the result

figure;
imagesc(lateral_axis, depth_axis, log10(abs(RawRF)));
colormap('gray'); % Set the colormap to grayscale
title('B-Mode Image with Log Compression');
xlabel('Lateral Position (mm)');
ylabel('Depth (mm)');

figure;
imagesc(lateral_axis, depth_axis, log10(abs(hilbert(RawRF))));
colormap('gray'); % Set the colormap to grayscale
title('B-Mode Image with Hilbert Transform and Log Compression');
xlabel('Lateral Position (mm)');
ylabel('Depth (mm)');
```

Figure 9. The MATLAB code used to generate images with log compression in Figures 10 & 11.

Figure 10 displays the image resulting from the logarithmically-compressed data (with no Hilbert Transform). The kidney, near field diffraction artifact, and multiple reflection artifact are also labelled. The near-field diffraction artifacts are identifiable as the diffraction patterns at the top of the image, nearest to the transducer. As the sound waves move further away from the transducer, their frequencies decrease and the diffraction patterns become more spaced out and more obviously concave (in the direction of original propagation). The multiple reflection artifact is identifiable by the wave patterns propagating in the opposite direction to the original propagation from the transducer (convex to the direction of original propagation).

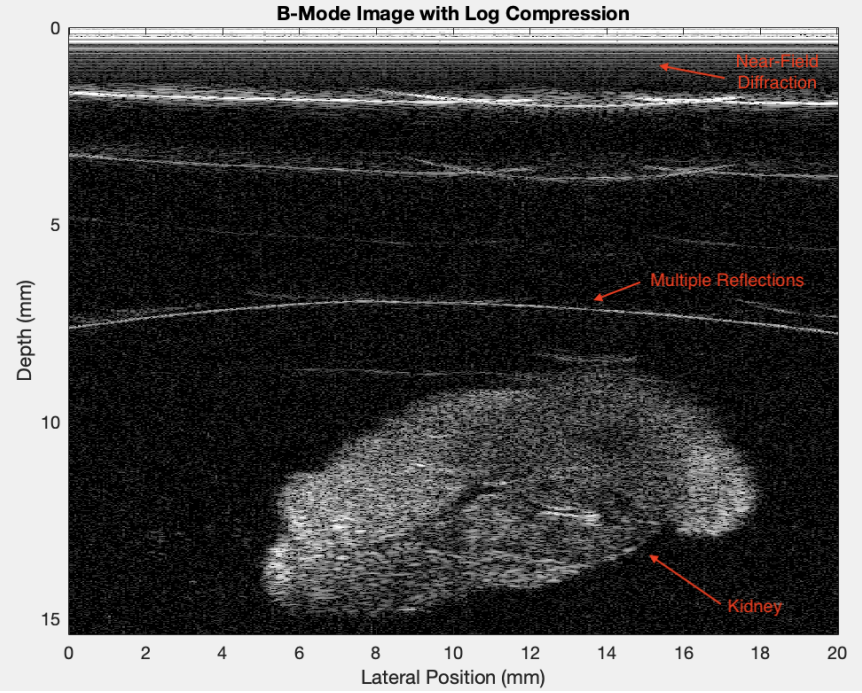

Figure 10. B-mode image after log compression is performed. The kidney, near field diffraction artifact, and multiple reflection artifact are labelled in red.

After log compression is performed, the previously lower amplitude signals in Figure 6 are now of a greater amplitude comparable to the signals that were already high in amplitude. This allows the lower amplitudes from different tissues within the kidney to be brighter and more apparent. Consequently, lower amplitudes in the background and diffraction and reflection artifacts are now brighter as well, contributing to more speckling noise.

The data which underwent Hilbert transformation was also logarithmically-compressed for comparison in Figure 11.

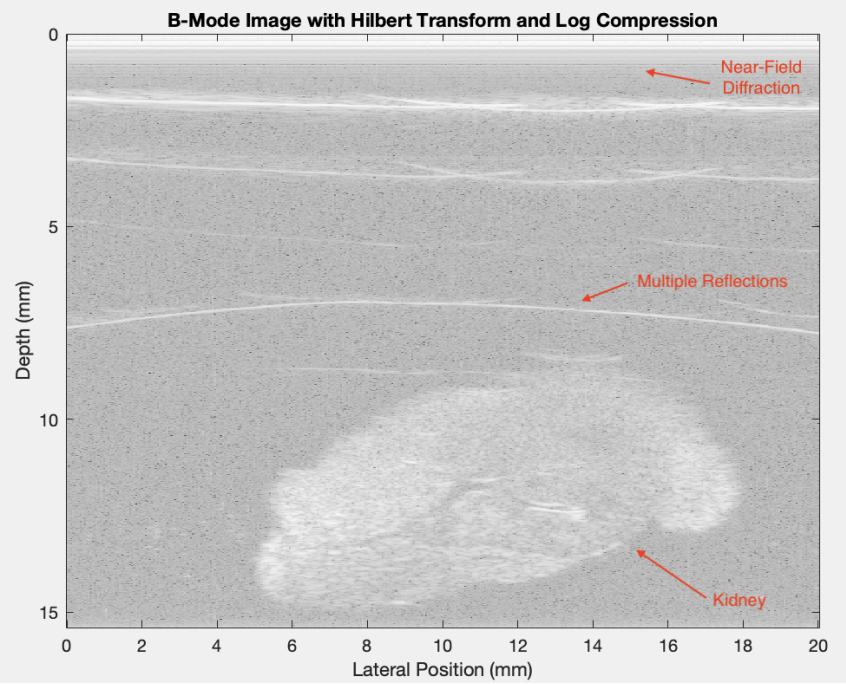

Figure 11. B-mode image with Hilbert transform and log compression. The kidney, near field diffraction artifact, and multiple reflection artifact are labelled in red.

Similar to the comparison made without the Hilbert transform in Figure 10, Figure 11 shows the previously lower amplitude signals in Figure 8 are now of a greater amplitude comparable to the signals that were already high in amplitude. The contrast-reducing effect of the Hilbert transform is displayed here as well, making the image overall brighter after log compression.

You will notice that log compression brings out low-intensity details as well as noise. Apply a suitable threshold to remove noise from the image, and repeat applying log compression. Plot the result and compare the image to that in the first part of Question 25.

To reduce resulting noise from the logarithmic compression processing, signals with amplitudes between -0.015 and 0.015 are rejected (made equal to 0). This threshold was selected principally by observing the baseline signal amplitude in Figure 4 and then verifying the resulting image had adequate resolution. These signals were rejected in an iterative process using a “for” loop shown in Figure 12.

```
image_rejection = RawRF;
threshold = 0.015;

for i= linspace(1,NumberOfDataPointsInFastTime,NumberOfDataPointsInFastTime)
    for j= linspace(1,NumberOfLines,NumberOfLines)
        if image_rejection(i,j) <= threshold & image_rejection(i,j) >= -threshold
            image_rejection(i,j) = 0;
        end
    end
end

figure()
figure7 = plot(time_vector, image_rejection(:,200));
title("Thresholded A-line (Line 200) with Noise Rejection (from 2 to 4 microseconds)")
xlabel("Time (microseconds)")
ylabel("Signal Intensity")
xlim([2 4])
ylim([-0.6,0.5])

figure()
figure8 = imagesc(lateral_axis, depth_axis,log10(abs(image_rejection)));
colormap(gray(256));
title("B-Mode Image with Log Compression and Noise Rejection")
xlabel("Lateral Position (mm)")
ylabel("Depth (mm)")
```

Figure 12. The MATLAB code used to generate the B-mode image with noise rejection (at an amplitude threshold of 0.015) and log compression in Figure 14 and visualise the processed signal in Figure 13.

For comparison with the original signal intensities in Figure 4, line 200 of RawRF with noise rejection is shown in Figure 13.

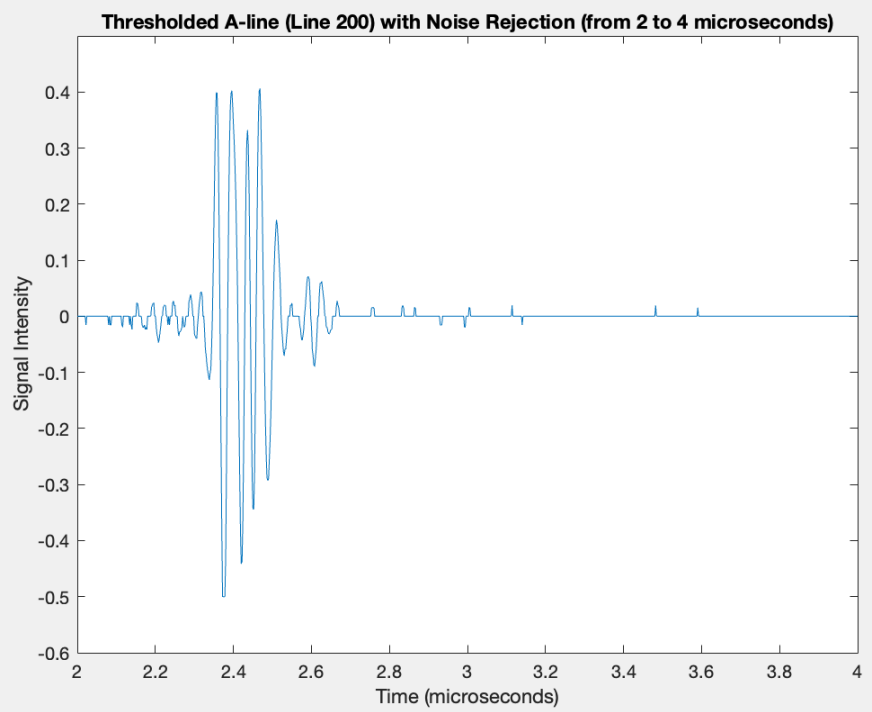

Figure 13. The signal intensity vs. time plot of a single A-line (line 200 of RawRF) after noise rejection (at an amplitude threshold of ±0.015) in the 2 to 4 µsec time interval.

The resulting noise thresholded and logarithmically-compressed B-mode image is displayed in Figure 14.

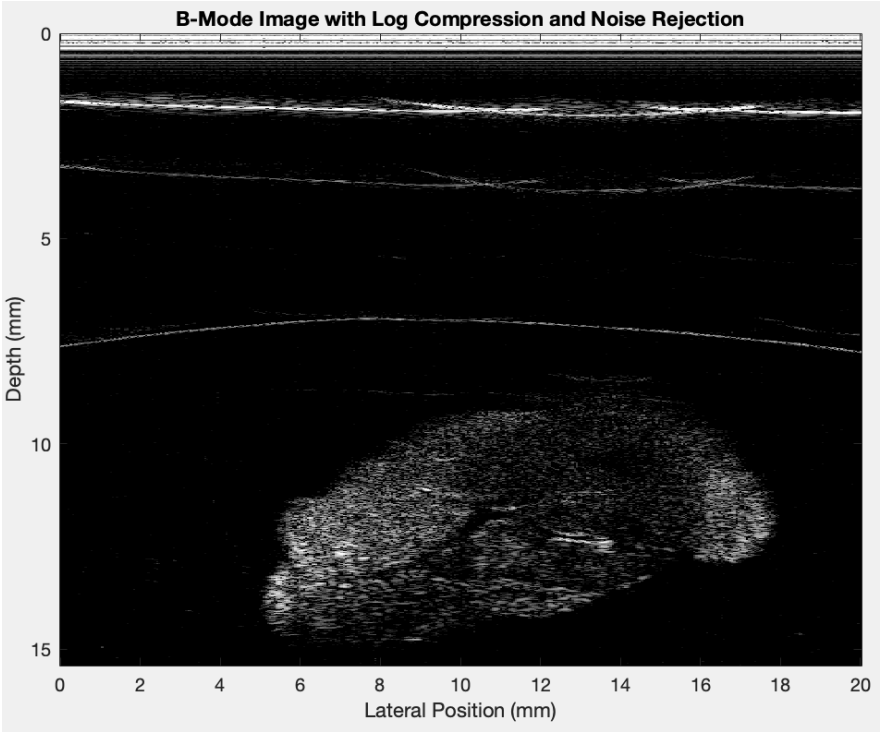

Figure 14. B-mode image with noise rejection (at an amplitude threshold of ±0.015) and logarithmic compression.

In comparison with Figure 10, the noise rejected image has significantly less noise speckling in the background and artifacts. This improved the overall contrast between the background and tissue structures in the kidney. By rejecting all signals with amplitudes between -0.015 and 0.015, the logarithmic compression of these zero-valued data points are undefined and therefore, are not amplified (brightened) in the image.

The data which underwent Hilbert transformation (envelope detection) was also corrected for noise rejection and then logarithmically-compressed for comparison as shown in Figure 15.

```
threshold = 0.015;

hilbert_rejected = hilbert(RawRF);
for i= linspace(1,NumberOfDataPointsInFastTime,NumberOfDataPointsInFastTime)
    for j= linspace(1,NumberOfLines,NumberOfLines)
        if hilbert_rejected(i,j) <= threshold & hilbert_rejected(i,j) >= -threshold
            hilbert_rejected(i,j) = 0;
        end
    end
end

figure()
figure9 = imagesc(lateral_axis, depth_axis,log10(abs(hilbert_rejected)));
colormap(gray(256));
title("B-Mode Image with Hilbert Transform, Noise Rejection, & Log Compression")
xlabel("Lateral Position (mm)")
ylabel("Depth (mm)")
```

Figure 15. The MATLAB code used to generate the B-mode image with envelope detection (through the Hilbert transform), noise rejection (at an amplitude threshold of 0.015), and log compression in Figure 16.

The resulting image from the above code is shown below in Figure 16.

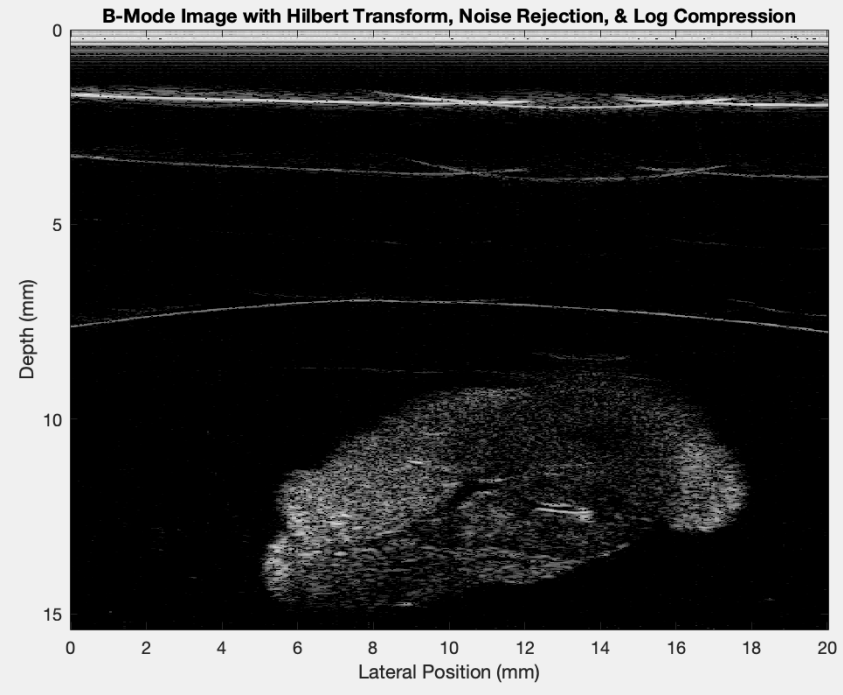

Figure 16. B-mode image with noise rejection (at an amplitude threshold of ±0.015), Hilbert transformation, and logarithmic compression.

Significant improvements to noise speckling and contrast were observed in this image above. The image is very similar to Figure 14 (without envelope detection), however the contrast and resolution appears to be slightly better without the Hilbert transform applied.

PART 2: PRECLINICAL ULTRASOUND IMAGING

1. ***What is the purpose of the ultrasound gel? What do you observe in the ultrasound image when air bubbles are trapped in the ultrasound gel?***

   The ultrasound gel serves as a conductive medium between the ultrasound transducer and the skin. By eliminating the transducer-air interface, the sound waves are transmitted more effectively into the body since they are not reflected from interfering air. Air bubbles in the ultrasound gel can result in artifacts in the ultrasound image due to air's high attenuation coefficient. The presence of air bubbles can reflect and scatter the waves when they reach an interface with the gel and air, disrupting the transmission of ultrasound waves and leading to poor image resolution and clarity. Due to the reflection of the ultrasound waves, air bubbles can appear as bright spots or lines in the image, covering underlying tissue and potentially leading to misinterpretation of the image.

2. ***Describe the image properties of the following structures: rib bones, liver, heart (left ventricle, left atrium, aorta, pulmonary artery, pulmonary vein). Explain both salient features and artifacts seen in each image.***

   The outline of rib bones appear as bright, hyperechoic lines, since bones are highly reflective to ultrasound waves due to their dense structure [2]. Since bones mostly reflect and minimally absorb sound waves, they create an "acoustic shadow" beneath them where ultrasound cannot penetrate, appearing as a dark area on the image.

   Liver typically has a uniform, medium-level echogenicity and appears as a relatively homogenous texture in ultrasound images, providing a good acoustic window for cardiac imaging [3]. It can exhibit enhancement artifacts where areas appear brighter due to the relatively uniform transmission of ultrasound waves throughout the liver.

   In the demonstration, the rat's heart wall appeared thicker than normal and the myocardium dominated the image in M-mode (as demonstrated by a large distance between the signals from the endocardium and epicardium), indicating the heart is unhealthy. If the ultrasound was performed on a younger rat, we would expect to observe larger left ventricle and thinner heart walls for more effective compression. In M-mode, this would appear as a larger overall cavity with a thinner myocardium (the myocardium's interfaces at the endocardium and epicardium, shown as lines in the image, would be closer together).

   In the ultrasound, the left ventricle (LV) appears as a more uniform shape and it can be visualised with a 2D view. LV is a muscular, dynamic chamber with wall movements synchronised with the cardiac cycle. In M-mode, motion artifacts may appear due to this movement of the heart walls. Characteristics of the LV such as wall thickness, volume, diameter, heart pumping power, and relaxation (diastole) quality can be identified from the signals of the M-mode image.

The left atrium (LA) appears posterior to and is a smaller cavity than the LV. The LA's shape is irregular as it curves around the heart and varies depending on the phase of the cardiac cycle, making it challenging to image and measure blood flow. The probe positioning and gain settings should be adjusted in order to avoid artifacts and to be able to capture the LA accurately. A 3D or 4D ultrasound may be conducted to collect more accurate measurements of all dimensions of the LA.

The aorta appears as a circular, anechoic (black appearing, with no internal echo due to minimal reflectivity) structure due to its blood filled nature, slightly right of the midline. "Mirror image" artifacts, where structures are duplicated on the opposite side of the reflector, can sometimes occur near strong reflectors like the aorta [4]. This type of artifact occurs when a primary beam reflection encounters another reflective surface, is then reflected back to the original surface of reflection, and then finally, reflected back to the transducer [5].

The pulmonary artery has a similar structure to the aorta, as it is tubular and blood filled and it appears near the aorta. The turbulent flow within the artery might create "colour bleed" when performing colour Doppler ultrasound.

The pulmonary vein, which brings oxygenated blood to the LA, appears as smaller anechoic circular structures, right under the aortic arch. Due to its size and position, the pulmonary vein can be challenging to image clearly and artifacts such as "colour bleed" (when performing colour Doppler) and "mirror image" may occur.

3. ***What is the purpose of time-gain compensation?***
   Time-gain compensation (TGC) is an important feature of ultrasound imaging used to compensate for signal attenuation occurring with depth of the subject being imaged. This process reduces the impact of wave attenuation with depth through increasing intensity (gain) of the received signal in proportion to the depth. Without TGC correction, ultrasound images would appear brighter near the surface and darker at deeper levels due to energy loss resulting from absorption and scattering of sound waves by tissue as sound travels deeper into the subject.

4. ***Now observe how colour Doppler is performed. What are the steps involved in obtaining a colour Doppler measurement? What do the colours represent? Specifically talk about colour intensity and direction/flow. Does the orientation/placement of the measurement probe matter?***
   Colour Doppler is an ultrasound technique that is used to visualise blood flow velocity (speed and direction) within the blood vessels by using pulse Doppler sound waves. Prior to performing colour Doppler, the transducer's and rat's position are adjusted, so that the heart can be visualised properly and the ultrasound gel is applied to its chest. Then, the ultrasound machine is switched to the colour Doppler mode, and gain is adjusted to increase contrast of the colours for optimal visualisation of the blood flow. In colour Doppler, oxygenated blood

is displayed as red, whereas de-oxygenated blood is displayed as blue. The direction of the transducer is adjusted to be the same way as the blood flow, so that red indicates flow towards the probe and blue indicates flow away from the probe. Since oxygenated and deoxygenated blood move in different directions, oxygenated blood can be distinguished from deoxygenated blood. The colour intensity corresponds to the velocity of blood flow. Brighter colours indicate higher velocity, while darker colours suggest slower blood flow. For accurate results, the ultrasound beam, and therefore transducer, should be near parallel to the blood flow direction, since according to the Doppler equation, the ideal Doppler angle of insonation is 0° [6]. To analyse blood flow and obtain measurements at a given time, the image may be frozen.

5. ***When assessing a long-axis view of the heart, what structures need to be visible and clearly outlined in the ultrasound image and Vevo-Lab in order to calculate accurate left ventricle dimensions? When assessing the pulmonary vein, what do you need to look for on the ultrasound image to be certain that the measurement is good? What three parameters can you extract on Vevo-Lab when using colour Doppler to assess the pulmonary vein? What do these parameters tell you about the overall function of the heart?***

When assessing a long-axis view of the heart, the left ventricle (LV) chamber itself should be fully visualised from the apex to the base. Endocardium, myocardium, and epicardium need to be present as they outline the LV and define the LV cavity volume & diameter. The interventricular septum (which separates the left and right ventricles), LV posterolateral wall, mitral valve (between left atrium and LV), and aortic valve (where aorta connects to the heart) should also be clearly visible to calculate LV dimensions, such as internal diameter, wall thickness and LV volume accurately [7].

When assessing the pulmonary vein using ultrasound, the point where the pulmonary vein enters the left atrium needs to be clearly visualised. The image should also be analysed for artifacts to ensure the image is stable and reproducible.

A, S, and D waveforms observed in the pulmonary vein during colour Doppler ultrasound helps assess the pulmonary vein function. The A-wave corresponds to atrial systole, and represents the blood flow into the left atrium from the pulmonary veins during atrial contraction. The S-wave corresponds to the ventricular systole, when the blood is pushed into the atrium due to the closure of the mitral valve, briefly reversing the flow. The D-wave corresponds to the ventricular diastole, where blood flows from the pulmonary veins into the left atrium and then into the relaxed left ventricle. Analysing the A-wave provides information about the quality of contraction of the left atrium and its ability to push blood into the left ventricle. A greater amplitude in A-wave may indicate a weaker myocardial contraction. The S and D waves provide information about the quality of blood flow during filling and draining of the LV, by assessing how well the ventricle contracts to push blood (systolic function) and how efficiently it relaxes and fills with blood during diastole.

References

[1] K. MacCallum, “Pros and cons of two popular ultrasound signal processing techniques,” StarFish Medical, https://starfishmedical.com/blog/pros-cons-two-popular-ultrasound-signal-processing-techniques/. (accessed Nov. 27, 2023).

[2] H. Qadi, J. Davidson, M. Trauer, and R. Beese, “Ultrasound of bone fractures,” Ultrasound (Leeds, England), https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7254943/#:~:text=Normal%20bone%20is%20made%20of,arrowheads)%20with%20no%20image%20beyond. (accessed Nov. 29, 2023).

[3] V. Dinh, “Cardiac Ultrasound (echocardiography) made easy: Step-by-step guide,” POCUS 101, https://www.pocus101.com/cardiac-ultrasound-echocardiography-made-easy-step-by-step-guide/#:~:text=Make%20sure%20to%20use%20the,downward%20pressure%20on%20the%20probe. (accessed Nov. 29, 2023).

[4] V. Dinh, “Aorta Ultrasound Made Easy: Step-by-step guide,” POCUS 101, https://www.pocus101.com/aorta-ultrasound-made-easy-step-by-step-guide/#:~:text=Identify%20the%20Aorta%2C%20a%20thick,is%20usually%20collapsible%20with%20respirations. (accessed Nov. 29, 2023).

[5] R. T, “Mirror Image Artifact: Radiology Reference Article,” Radiopaedia, https://radiopaedia.org/articles/mirror-image-artifact (accessed Nov. 29, 2023).

[6] “Doppler effect,” Doppler Effect - an overview | ScienceDirect Topics, https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/doppler-effect#:~:text=Ultrasound%2C%20Carotid&text=According%20to%20the%20Doppler%20equation,decreases%20and%20approaches%200%C2%B0. (accessed Nov. 29, 2023).

[7] “3.2.2 size of the left ventricle,” 123 Sonography, https://123sonography.com/ebook/size-left-ventricle (accessed Nov. 28, 2023).