



**HWI HAMBURG**  
HOCHSCHULÜBERGREIFENDER STUDIENGANG  
WIRTSCHAFTSINGENIEURWESEN HAMBURG



Hochschule für Angewandte  
Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*



 **Fraunhofer**  
MEVIS

# MASTER'S THESIS

**Implementing a Low-Cost, Open-Source Raspberry Pi Development  
Environment with a Single-Element Ultrasonic Transducer**

**Technical Capabilities, Future Research Applications & Implications for the Health Market**

*Hamburg/Bremen, 2<sup>nd</sup> December 2019*

## **Abstract**

**Topic** - Simple and portable ultrasound devices are considered to enhance medical care in the future. The open-source UnOrick ultrasound board forms the basis for a simple low-cost (540 EUR) ultrasound system. It is supplemented by an open-source software Python-environment. In combination with a single-element transducer and controlled by a Raspberry Pi, the aim of this project is to implement and characterise such a system.

**Scope of Research** - This master's project explores the technical capabilities of this simple low-cost ultrasound system (LCUS) in three areas of development: Hardware Set Up, Software and Experimental Tests. Available open-source software is reviewed, missing methods for signal processing and image reconstruction are developed and a graphical user interface is implemented. Furthermore, the implications for the health care market of such a frugal device are explored. Finally, future research opportunities are proposed.

**Main Findings** - The implemented system allows to acquire 2D ultrasound images pulsing at a frequency of 1MHz and 72V and scan an angle of up to 60 degrees. A band-pass FFT- Filter and averaging should be applied to improve the scans. Image resolution is significantly improved by measuring the PSF and deconvolving the images by applying Wiener-Filtering. Being considerably less costly than comparable ultrasound devices on the market and highly adaptable due to the open-source nature, the LCUS shows high potential for future research applications.

**Future Research** - Research fields Open Innovation, Frugal Innovation and Modularity underline the utility of the LCUS as a product. Future integration of artificial intelligence methods and or the used of aberration masks could ameliorate the achievable image resolution.

**Keywords:** *Low-Cost Ultrasound System, Frugal Innovation, Open-Source Ultrasound Hardware, UnOrick Board, Hand-Held Device, Raspberry Pi Python 3 Development Environment, 2D Image Processing, Single-Element Transducer, Student Research Project in Ultrasound Imaging, Deconvolution*

## Contents

0.1	Table of Figures .....	IV
0.2	Explanations .....	VI
0.3	Table of Abbreviations.....	VI
1.	Introduction .....	1
1.1	Advantages of Ultrasound as an Imaging Modality in Medicine .....	1
1.2	Scope of Research and Formulation of Research Questions .....	4
1.3	Theoretical Underpinning.....	6
1.3.1	The Basic Physical Principles of Ultrasound Imaging.....	6
1.3.2	Ultrasound Field of Piston Transducer .....	8
1.3.3	Electronic Components of a Basic Ultrasound System.....	13
2.	Materials and Methods.....	14
2.1	Hardware Architecture of the Low-Cost Ultrasound System.....	15
2.1.1	Raspberry Pi .....	15
2.1.2	UnOrick Board.....	16
2.1.3	Transducer.....	17
2.1.4	Measuring Devices: Oscilloscope, Network Analyser and Signal Generator.....	17
2.1.5	Phantoms, Transducer Holder & 3D Printer.....	18
2.1.6	Experimental Set Up.....	19
2.1.7	Further Equipment.....	20
2.1.8	Estimated Cost of Low-Cost Ultrasound System .....	20
2.2	Software Architecture of the Low-Cost Ultrasound System .....	21
2.2.1	Python .....	21
2.2.2	Octave .....	21
2.2.3	MeVisLab .....	21
2.2.4	Software Development .....	22
2.3	Methods of Experimental Testing .....	31
3.	Results.....	36
3.1	Determination of the Transducer's Resonance Frequency .....	36
3.2	Simulation of the Transducer's Ultrasound Field .....	37
3.3	Analysis of UnOrick Board's Sampling Frequency and Voltage .....	38
3.4	Signal Fast Fourier Transformation Frequency Spectrum Analysis .....	41
3.5	Application of Image Averaging.....	44
3.6	Determination of Axial Image Resolution .....	46
3.7	Determination of Lateral Image Resolution .....	47
4.	Discussion .....	52
4.1	What are the Technical Capabilities of the Low-Cost Ultrasound System?.....	52
4.2	Which Implications for the Health Care Market can be derived? .....	55
4.3	What are Future Research Applications of the System? .....	58
5.	Conclusion.....	61
6.	Bibliography .....	62
7.	Annexe .....	66

## 0.1 Table of Figures

### *Chapter 1: "Introduction"*

Figure 1.1:	Comparison of Imaging Modalities: Ultrasound, CT and MRI .....	2
Figure 1.2:	The Piezo-Electric Effect Explained .....	6
Figure 1.3:	Pulse Generation with a Piezo-Electric Crystal .....	6
Figure 1.4:	Pulse Transmission, Echo Reflection and Received Signal .....	7
Figure 1.5:	The Ultrasound Field of a Low Frequency Piston Transducer (<2MHz) .....	8
Figure 1.6:	The Rayleigh Criterion applied to Utrasound Signals .....	9
Figure 1.7:	Exemplary Experimental Set Up and its theoretical Echo Signal Response .....	10
Figure 1.8:	Transducer's Pressure Field .....	10
Figure 1.9:	Rotating Pressure Field of Single-Element Transducer (simplified).....	11
Figure 1.10:	Schematic of Image Convolution and Deconvolution Process .....	12
Figure 1.11:	Block Diagramme of Basic Ultrasound System .....	13

### *Chapter 2: "Materials & Methods"*

Figure 2.1:	Methodology of Implementing the Low-Cost Ultrasound System (LCUS) .....	14
Figure 2.2:	Hardware Architecture of the Low-Cost Ultrasound System .....	15
Figure 2.3:	UnOrick Board's Interfaces .....	16
Figure 2.4:	The Ultrasound Pulse generated by UnOrick Board and its Pulse Param. ....	17
Figure 2.5:	3D-printed Transducer Holder & Step Phantoms, Metal Plate Phantom .....	18
Figure 2.6:	Experimental Set Up, RP, UnOrick Board and Step Motor .....	19
Figure 2.7:	Image Acquisition Set Up and Image Axes (Image Acquisition Routine) .....	20
Figure 2.8:	Costs of Low-Cost Ultrasound Device .....	20
Figure 2.9:	The Software Architecture .....	22
Figure 2.10:	Reviewing and Selecting the Functions of the UnOrick-Software .....	23
Figure 2.11:	Implemented Signal Processing and Image Reconstruction Process .....	25
Figure 2.12:	Us_experiment.py – Acquiring an Image with the LCUS .....	27
Figure 2.13:	Tab1: Acquisition Parameters Interface .....	28
Figure 2.14:	Tab 2: Web Socket Connection Interface .....	29
Figure 2.15:	Tab 3: Signal Processing and Image Reconstruction Interface .....	30
Figure 2.16:	Axial Resolution Test: Step Phantom 2 and its Runtime Distance .....	34
Figure 2.17:	Experimental Set Up for Determination of Point Spread Function .....	35

*Chapter 3: "Results"*

Figure 3.1:	Network Analyser Test S-11 Parameters of the Single-Element Transducer .....	36
Figure 3.2:	Results of MeVisLab Ultrasound Field Simulation .....	37
Figure 3.3:	Results of Voltage Test.....	39
Figure 3.4:	Voltage Intensity Plot of Single Acquisition Line .....	40
Figure 3.5:	Fast Fourier Spectrum Analysis of One Recorded Signal Line .....	42
Figure 3.6:	Harmonic Image of Fig. 3.3 , FFT-Filtered Image & Difference Image .....	43
Figure 3.7:	Envelopes of Central Signal Lines of 25 Acquired Images & Averaged Signal ..	44
Figure 3.8:	Results of Averaging .....	45
Figure 3.9:	Recorded Echo Signal of Axial Resolution Test Phantom.....	46
Figure 3.10:	Flat Acquired Images of a 3mm Point Source in 8 Z-Positions .....	47
Figure 3.11:	Gaussian Fits of Point Spread Function .....	48
Figure 3.12:	Deconvolved "Flat 2D images" .....	49
Figure 3.13:	Zoom on Point Source Object Before and After the Deconvolution .....	50
Figure 3.14:	Sum of 5 Deconvolved Reconstructed "True 2D Images" .....	51

*Chapter 4: "Discussion"*

Figure 4.1:	Comparison of Properties commonly used HHD and LCUS .....	55
Figure 4.2:	Phase Encoding Masks .....	60

## 0.2 Explanations

- All mentioned Python methods are written in *Courier New* font in the following text.  
Example: `write_fpga()` is a method called upon to write commands to the FPGA Chip.
- All Python Scripts are expressed in ***Calibri Italics bold*** appended by a `.py` formatting.  
Example: In the ***us\_experiment.py*** all parameters are specified.

## 0.3 Table of Abbreviations

2D/ 3D	2-Dimensional / 3-Dimensional
5G	Fifth Generation (Internet)
AI	Artificial Intelligence
CT	Computer Tomography
DAC/ADC	Digital to Analogue Converter/ Analogue to Digital Converter
EUR	Euro
EFSUMB	European Federation of Societies for Ultrasound in Medicine and Biology
FFT	Fast Fourier Transformation
FPGA	Field Programmable Gateway Array
FI	Frugal Innovation
GPIO	General Purpose Input Output
GUI	Graphical User Interface
HHD	Hand-Held (Portable) Ultrasound Device
IEC	International Electrotechnical Commission
IP	Intellectual Property
LCUS	Low-Cost Ultrasound System (the System characterised in this Project)
LMIC	Low and Middle Income Countries
LSQR	Least Squares Root Algorithm
MRI	Magnetic Resonance Tomography
PSF	Point Spread Function
Python Packages	Numpy, Scipy, Matplotlib
RP	Raspberry Pi
Savgol-Filter	Savitzky-Golay Filter
SMA	Sub-Miniature Version A (Interface)
SNR	Signal-to-Noise Ratio
Un0rick Board	Open-Source Ultrasound Board used in this Project
USB	Universal Serial Bus
TGC	Time Gain Compensation
WHO	World Health Organisation

## **1. Introduction**

The discovery of piezo-electricity by physicists Jaques and Pierre Curie in 1880 paved the way for industrially generating high frequency ultrasound waves. These were initially used in naval sonar systems to detect objects in the ocean. The application of ultrasound to medical imaging, however, took another 80 years until commercial scanners were developed. In the 1960ies, Australian researchers Robinson and Kossoff introduced a method to scan foetuses in obstetric examinations. Medical Ultrasound (also called “Ultrasonography” or “Diagnostic Sonography”), is widely applicable to scan various parts of the human (and non-human) body to generate images (or videos) of the region of interest in order to diagnose and treat pathologies.

### **1.1 Advantages of Ultrasound as an Imaging Modality in Medicine**

In 2017, the increase in German health expenditures amounted to 4.7 % compared to the year before (Statistisches Bundesamt, GENESIS online platform) [1], continuing the trend of rising costs of the health care system. This incline is expected to proceed due to demographic change and other social factors. Every year, 3% of the budget for health costs in Germany is spent on “*diagnostic imaging services*”. These include - amongst other imaging modalities in radiology – Magnetic Resonance Tomography (MRI), Computer Tomography (CT) and Ultrasound examinations [1]. Recently, over a span of 10 years (2007-2017), the budget spent on diagnostic imaging examinations in German clinics, practices and other medical service providers rose by 60 %, amounting to 10.4 Billion EUR in 2017 [1]. On the basis of this development, medical imaging devices and examinations are required which are less cost-intensive than existing ones to ensure the standard of medical care in the future.

In the year 2016, 12 million *diagnostic imaging examinations* were conducted in Germany (evaluation of DRG report 2016) [2]. For these medical consultations, the three most commonly used imaging modalities are CT with 5.8 Mio., MRI with 2 Mio. and Ultrasound with 1.2 Mio. [2] examinations per year. To give an overview of the three imaging modalities, Figure 1.1 compares them - amongst other criteria - concerning the *safety* and *cost* of use and device *portability*.

Using ionising X-ray to scan the body, CT poses health risks for patients. MRI as well as Ultrasound scans are risk free. Using MRI technology, however, comes at a higher cost than ultrasound imaging. In the US an MRI examination costs on average 2.300 EUR (2.600 USD) (analysed medicare programme data, Times Magazine, 2019) [4]. Medical staff training and initial investment in the technology are vast. A state-of-the-art 3 Tesla MRI machine costs roughly 2.5 Mio. EUR. Additionally, the technology requires building special MRI suites, rooms to house the machines, and its development of hard- and software is very resource intensive.

<b>Modality</b>	<b>Ultrasound</b>	<b>CT</b>	<b>MRI</b>
What is imaged	Mechanical properties	Tissue absorption	Biochemistry ( $T_1$ and $T_2$ )
Access	Small windows adequate	Circumferential Around body	Circumferential Around body
Spatial resolution	Frequency and axially dependent 0.3–3 mm	~1 mm	~1 mm
Penetration	Frequency dependent 3–25 cm	Excellent	Excellent
Safety	Very good	Ionizing radiation	Very good
Speed	100 frames/sec	$\frac{1}{2}$ minute to minutes	10 frames/sec
Cost	\$	\$\$\$\$	\$\$\$\$\$\$\$
Portability	Excellent	Poor	Poor

**Figure 1.1: Comparison of Imaging Modalities: Ultrasound, CT and MRI (adapted from [3])**

Ultrasound examinations, on the other hand, are estimated to cost between 40 and 100 EUR (DRG report 2008) [5]. Even conservatively comparing the cost of using these imaging modalities, MRI scans are 20 times costlier than ultrasound examinations. Another advantage of ultrasound imaging is that systems are portable (high-end systems as used in clinics are mounted on a trolley to be moved to the patient's bedside). Smaller systems referred as "Hand-Held Devices" (HHD), which have the dimensions of a laptop computer or are even smartphone-sized, exist. These portable HHD are subject to recent research in the field of medical ultrasound imaging.

#### *Hand-Held (Portable) Ultrasound Devices*

In an extensive recent systematic review on the comparison of hand-held portable ultrasound devices to high-end Ultrasound systems commonly used in clinics and practices, Rykke et al. [6] conclude that HHD hold the potential to become a "game-changing" technology concerning the availability and cost of ultrasound examinations. The review identifies 16 promising studies conducted between 2012 and 2018 in chest and lungs, abdominal, urological, gynaecological and vascular ultrasound examinations, showing strong results in specific applications such as hydrophonephrosis detection and screening for abdominal aortic aneurysms/ measuring the aortic diameter with hand-held portable devices. Another review of pocket-sized imaging devices, conducted by Galusko et al. [7] confirms the diagnostic utility in echocardiography if handled by an experienced professional. These results are underlined by the discussion of further potential clinical applications of such devices in a strategy paper (updated 2018) of the European Federation of Societies for Ultrasound in Medicine and Biology (EFSUMB) [8] in abdominal, lung and paediatric ultrasound as well as for the *training of medical students*. Moreover, Becker et al. [9] found that using a hand-held portable ultrasound device in an examination when a high-end ultrasound system is not available might have a positive impact on clinical management in 70% of the cases during natural disasters, when a high frequency of patient examinations is

conducted. This was investigated in low- and middle-income countries (LMIC). As the use of ultrasound systems require significantly less training and are less costly than other imaging modalities, the World Health Organisation (WHO) recognises and stresses the advantages of using medical ultrasound in LMIC [10]. In these regions, hand-held devices are commonly used for echocardiography in rural areas, in 90% of examinations obviating the need for a comprehensive echocardiogramme [9] or other imaging modalities for confirmation.

In the literature, a majority of studies use the General Electric (GE) VScan system (Price: approx. 3.900 EUR [11] as reference HHD. It is therefore used for a short comparison in the “Discussion” of this master’s thesis in Chapter 4.

#### *Future of Ultrasound Image Analysis*

Another advantage of ultrasound as an imaging modality originates from the emergence of faster computing power and new procession methods such as parallelisation. These are considered to advance ultrasound imaging in the future. Paired with 5G-internet technology and cloud-based image analysis, ultrasound examinations could be made available in remote locations, where the interpretation of the results is performed in real-time by a spatially distant physician [6]. Furthermore, Artificial Intelligence (AI) technology is considered to bear the potential of speeding up scanning and enhancing ultrasound image quality by partly atomising and supporting the diagnostic process so that hand-held devices could catch up with image quality achieved by commonly used ultrasound systems [6] and even other imaging modalities such as MRI.

#### *Postulating the Motivation for Research in this field*

Many stake-holders have declared that medical ultrasound imaging could be seen as a solution to encounter rising costs in the German health care industry and enhance the provision of medical health care in LMIC. Compared to other existing imaging modalities, it excels by not using ionising radiation and being inexpensive with respect to the technology itself as well as training and other peripheral factors. This motivates further research in the field. The development of small portable ultrasound devices coupled with computing power and AI approaches might present a “game-changing” technology and should be pursued further. The investigation of such a system is therefore subject of research in this master’s project.

## **1.2 Scope of Research and Formulation of Research Questions**

Advantages and advances of ultrasound systems – especially portable devices – in mind, the aim of this master’s project is to implement and characterise a simple low-cost ultrasound system (LCUS). With the help of open-sourced hard- and software components, it is achievable to build such a frugal inexpensive ultrasound device. To describe the work which this master’s project is based on, two open-source projects require introduction. They produced the main hardware components used – an open-source ultrasound board “UnOrick” and the mini-computer “Raspberry Pi”.

### *Rationale of UnOrick Project*

The ultrasound board utilised was designed by an international team of contributors led by Luc Jonveaux [12] who aimed at developing an open-source ultrasound device which would be considerably less costly than existing commercial ultrasound scanners on the market [12]. Whilst there are multiple open-source ultrasound software projects (a summary is given in [12]) working on image processing, the developers identified a lack of *open-source hardware* available to researchers and the broad public [12]. Components used in the development of the UnOrick board were developed in previous open-source projects [13] (echomods projects, Jonveaux et al. [12]). Taylor, Jonveaux and Caskey [14] propose the use of such a frugal imaging system as medical device in the developing world as it is inexpensive and portable. In their initial test set ups, an Arduino Uno board was used for micro-controlling. For this master’s project, a Raspberry Pi (RP) mini-computer is used to control the ultrasound system and conduct ultrasound image processing.

### *Raspberry Pi Foundation’s Philosophy*

Therefore, another central component of the low-cost ultrasound system is the mini-computer Raspberry Pi. The Cambridge-based Raspberry Pi Foundation sets its mission to put “power of computing and digital making into the hand of people all over the world” (Raspberry Pi Foundation Strategy paper 2018-20, [15]) in order to “solve problems that matter to them”. The Foundation emphasises its open and collaborative approach to developing the low-cost mini-computer. They specifically aim at providing affordable and free software for educational and research purposes [15]. The Raspberry Pi is embedded in a large support infrastructure online. With its price of just under 40 EUR, a Raspberry Pi control unit therefore aligns with this master’s project goal to build a *low-cost* ultrasound system.

### *Researched System*

Additional to these two hardware components, the system is completed by a single piezo-element ultrasonic transducer and other inexpensive components. Free open-source software Packages Python 3, GNU Octave and MeVisLab are used to implement the system under investigation.

### *Formulation of Research Questions*

Implementing and characterising the *Low-Cost, Open-Source Raspberry Pi Development Environment with a Single-Element Ultrasonic Transducer* spurs three domains of research which are interesting to explore:

- RQ1: What are the system's technical capabilities?*
- RQ2: Which implications for the health care market can be derived?*
- RQ3: What are future research applications of the system?*

### *Novelty of the Work*

Taylor, Jonveaux and Caskey [14] have successfully implemented such a system using an Arduino-board for controlling. Additionally, the Un0rick team has so far published several tests on their GitHub repository using the RP for control in which single ultrasound lines were acquired, FFT-filtered and plotted over time.

In this master's project, their software is extended by new signal processing and image reconstruction routines allowing to render 2D ultrasound images and apply more sophisticated methods for imaging such as the deconvolution of images for resolution improvement. Experimental tests are conducted to characterise the technical properties of the system. This includes the exploration of averaging behaviour, measuring the axial resolution and the point spread function of the system. Moreover, the Fraunhofer MEVIS Software "MeVisLab" is used to develop a graphical user interface (GUI) which is used on a laptop PC establishing a web-socket connection over Ethernet. Finally, the characterised system is mapped into recent fields of research in innovation and technology management and a future outlook is given on further research opportunities of the system. Here, especially combining the LCUS system with the concept of phase encoding masks is proposed.

### *Structure*

All hardware and software components are described in detail in Chapter 2, "Materials and Methods". While the hardware is explained in Chapter 2.1, "Hardware Architecture", available Un0rick software for the board is reviewed and implantation of missing functionalities described in Chapter 2.2, "Software Architecture". The conducted experimental tests are outlined in Chapter 2.3, "Methods of Experimental Testing". The results of these experimental test are shown and analysed in Chapter 3, "Results". In the final part of this thesis in Chapter 4, "Discussion", the formulated research questions are discussed. This includes a future outlook on further research opportunities. The "Conclusion", Chapter 5, summarises main findings of this project.

## 1.3 Theoretical Underpinning

How does medical ultrasound work? Ultrasound imaging is based on several physical and electronic principles. The imaging modality also exploits classical methods of signal processing. These need to be sketched briefly to explain the context of the project.

### 1.3.1 The Basic Physical Principles of Ultrasound Imaging

The physical principle which ultrasound is based on, is the generation of pressure waves in a medium. Ultrasonic (“ultra” is borrowed from Latin, meaning “beyond”) waves are pressure waves oscillating at a frequency higher than discernible by the human ear (16kHz- 18kHz [17]). The frequencies used in medical ultrasound range from 1MHz to 15MHz depending on the medical application, propagating through the human body at an average speed of sound of  $c_{av} = 1540 \text{ m/s}$  [3].

*Piezo-Electric Effect.* The mechanical property which allows to generate pressure waves is found in piezo-electric crystals (ceramics). When a voltage is applied to them, the crystals are contracting. This phenomenon describes the direct piezo-electric effect within the crystal (as seen in Figure 1.2). The reverse effect is also valid. When an ultrasound pressure (sound) wave hits the surface of the element and compresses the crystal, a voltage is induced and an electronic signal can be recorded.

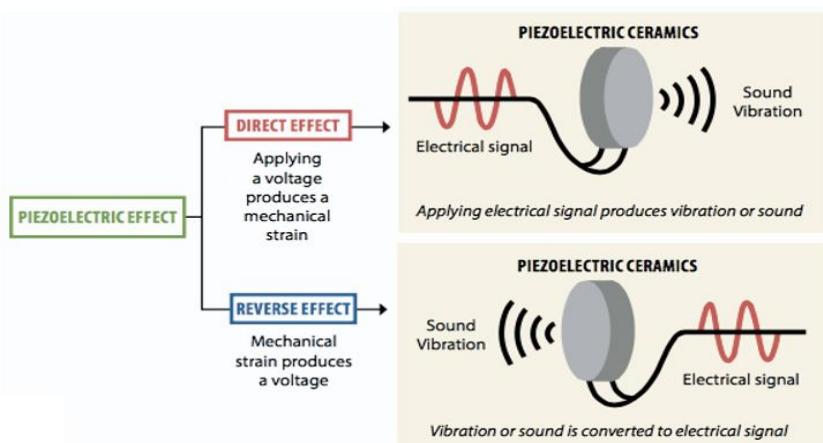


Figure 1.2: The Piezo-Electric Effect Explained [17, adapted]

*The Ultrasound Pulse and Transducer.* By periodically applying a voltage to the piezo-element (see Figure 1.3), a single or train of pressure waves is generated and emitted into a medium. The stimulus package emitted is called a pulse. Its propagation direction is orthogonal to the piezo-

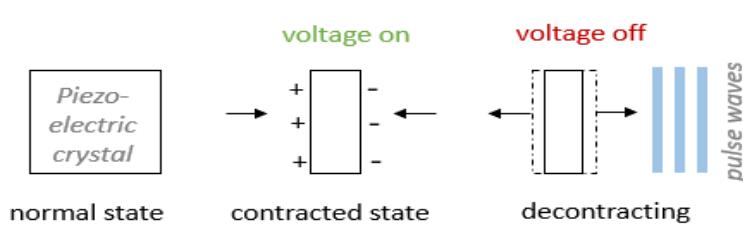
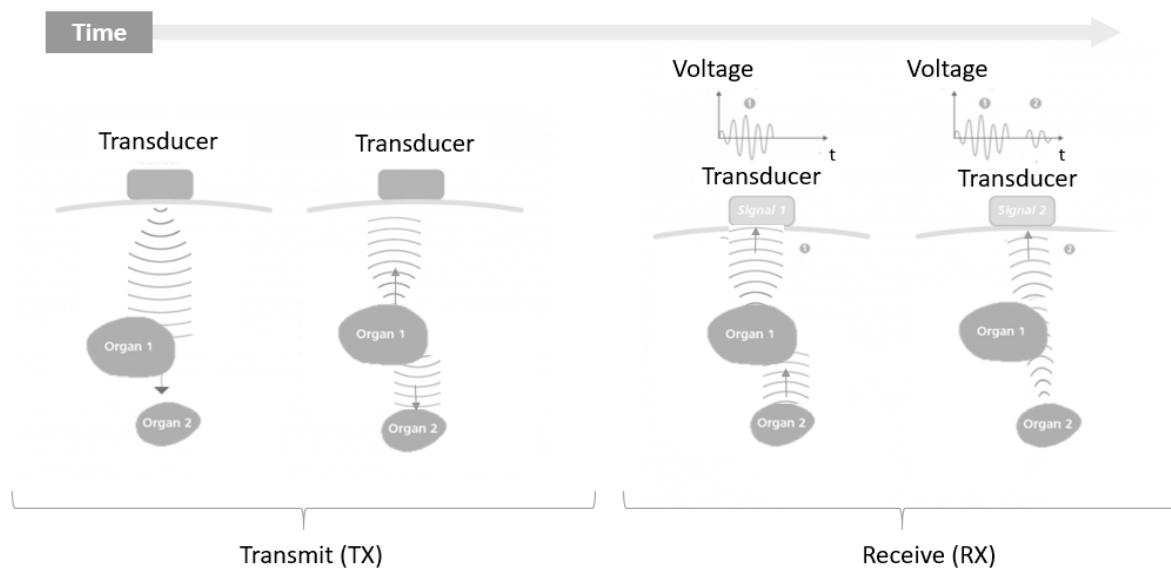


Figure 1.3: Pulse Generation with a Piezo-Electric Crystal

element's surface. The pulse generating component of an ultrasound system carrying piezo-elements is called a transducer, emitting pulses and records echo signals.

*Transmit and Receive Signal.* When the emitted pulse reaches tissue surfaces in the human body (e.g. organs), the pressure wave is reflected at interfaces between tissue layers of different acoustic impedance (linked to the density of the medium). The strength of the echo is proportional to the difference in impedance [17]. If there is no impedance difference between tissue interfaces, no echo is reflected. Homogenous fluids such as water are thus seen as echo free structures [17]. Each human tissue type, both normal and diseased, has a characteristic ultrasound appearance. Fluid-filled structures (blood vessels, gallbladder, bladder) typically produce no signal. Strong reflectors, such as bone, reflect most of the signal [17]. The echo waves travel back to eventually hit the transducer's surface and then acts as a receiver. Figure 1.4 depicts the transmission, reflection and reception process. Due to the inverse piezo-electric effect, a voltage is induced in the transducer. The ultrasound system records this resulting signal over time.



**Figure 1.4: Pulse Transmission, Echo Reflection and Received Signal [16, adapted]**

Runtime is counted starting from the moment of pulse emission until the echo signal is received. The temporal information can be translated into spatial information. Knowing the speed of sound in the respective medium, the conversion is calculated [17] as follows:

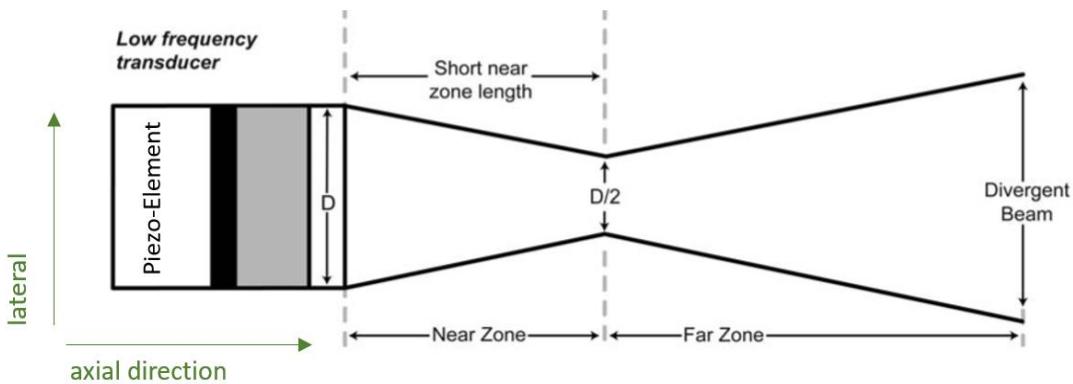
$$\text{distance } (d)[\text{m}] = \text{speed of sound } (c) \left[ \frac{\text{m}}{\text{s}} \right] * \text{runtime} \frac{(t)[\text{s}]}{2} \quad (1) *$$

The acquisition of a single line with a single-piezo element along a 1-dimensional axis is called A-mode scanning [17]. In order to add more information to render 2D images (B-Mode) as widely done in modern ultrasound imaging, multiple piezo-elements are used. Another option to achieve 2D scanning is to rotate the single-element transducer and acquiring single lines in rapid succession. The recorded electric signals require conventional processing methods applied in image processing. Ultrasound images are typically presented in grey-value images.

\* This equation is later used for the 2D reconstruction algorithm “Flat 2D Images” (see Annex)

### 1.3.2 Ultrasound Field of Piston Transducer

A circular piezo-element (piston) transducer, such as the one used in this project, forms an ultrasound field of a particular shape when emitting a pulse in axial direction. Near the transducer, the beam width is approximately equal to the width of the transducer [18]. It then converges in the near-field into a natural focus point of half the transducer width after which it then diverges infinitely (see Figure 1.5) in its far-field zone (Fraunhofer Zone).



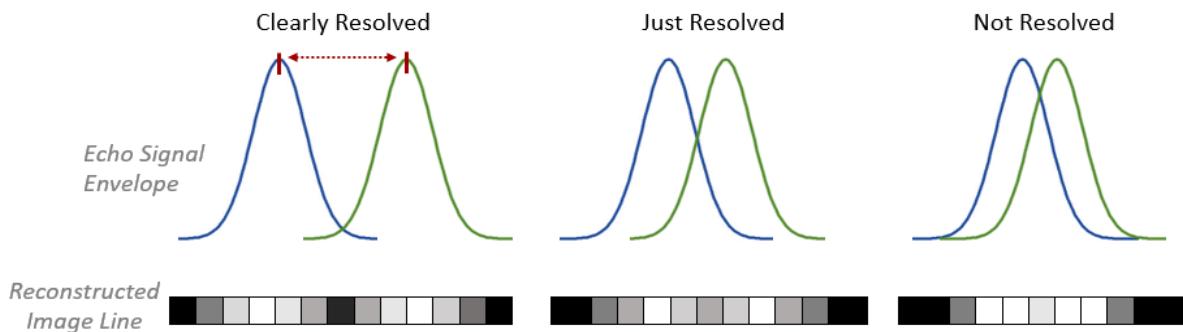
**Figure 1.5:** The Ultrasound Field of a Low Frequency Piston Transducer (<2MHz) [13]

#### Axial Resolution of an Ultrasound System

The minimal distance distinguishable between two reflecting objects, i.e. resolved in an ultrasound image, orthogonal to the direction of the ultrasound beam [18] describes the axial resolution of the system. It can theoretically be approximated by equation (2) [19] (approximation of the Rayleigh criterion). The spatial pulse length - a quotient of the number of cycles in a pulse and the frequency [13] - is weighted by a factor 0.77.

$$\text{Axial Resolution [mm]} = \frac{0.77 * N_{cycles}}{f[\text{MHz}]} \quad (2)$$

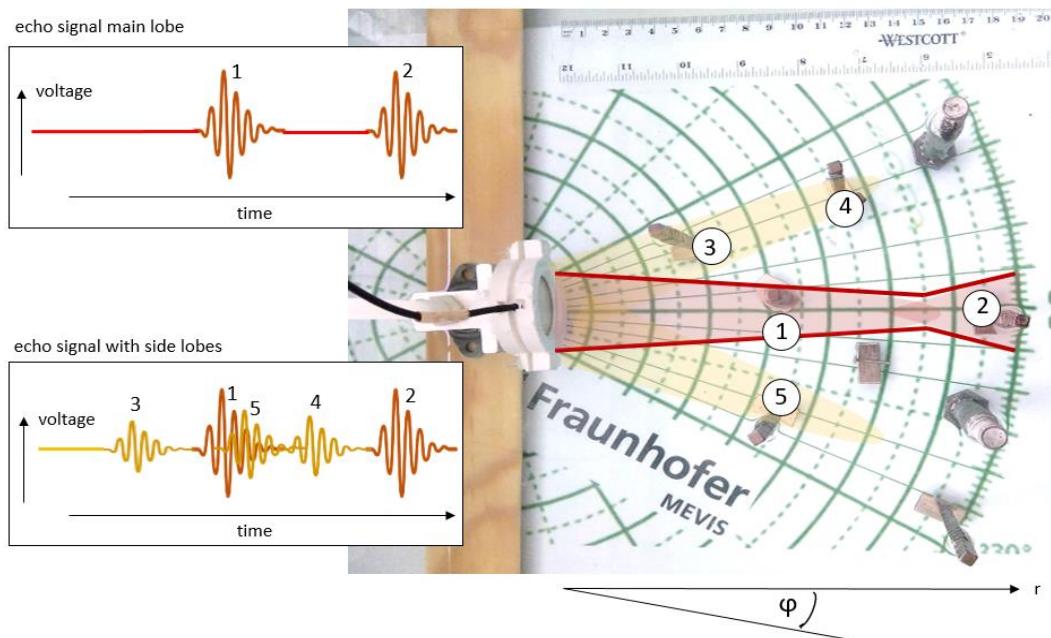
Therefore, the axial resolution of the ultrasound signal can alternatively be described as the threshold distance between two echo signal peaks where the viewer of reconstructed image (grey values) can just distinguish between two objects. This is shown in Figure 1.6. From equation (2), it can be deduced that a higher pulse frequency increases the axial resolution.



**Figure 1.6: The Rayleigh Criterion applied to Ultrasound Signals [20]**

*Lateral Resolution of an Ultrasound System.* Lateral resolution, analogous to the axial resolution, is the minimum distance that can be distinguished between two reflectors located on an axis perpendicular to the scan direction of the ultrasound beam [17]. If the width of the ultrasound beam is low, the lateral resolution is high and vice versa [17]. Since the width of the ultrasound beam varies over distance from the transducer, the lateral resolution varies along its axis [17]. At the natural focus point, the theoretical lateral resolution is half of the transducer's diameter (Figure 1.5). An increase in transducer diameter leads to a proportional deterioration on lateral resolution.

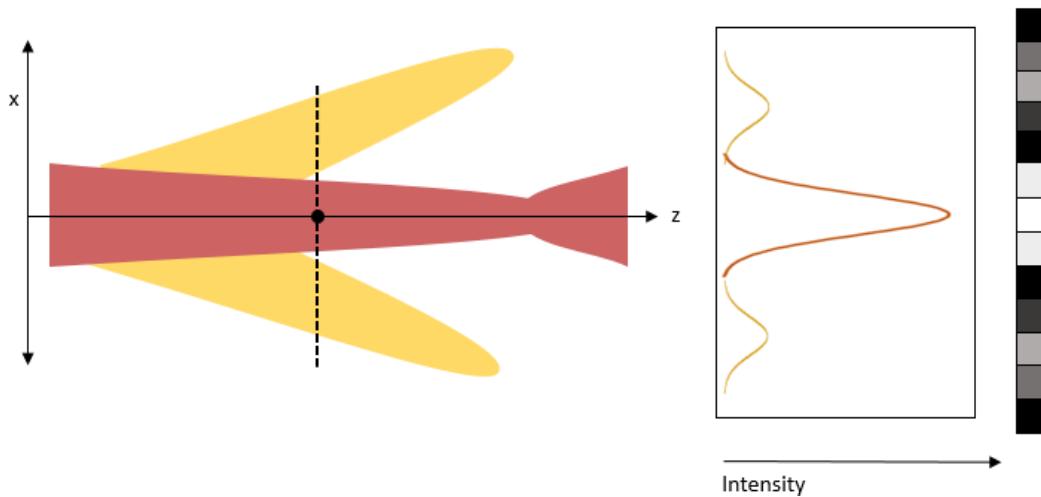
*Side Lobes of Ultrasound Beam.* Due to additional mode vibrations of the radial piezo - electric crystal, undesired secondary (side) lobes originate from the transducer in a different direction than the primary (main) beam [21]. These occur symmetrically in an angle of maximally 90 degrees from it. The intensity of side lobes is decreasing with higher order and their intensity magnitude is considerably lower compared to the main beam [22]. This means in some applications (such as liver scans) they are negligible [21]. Ignoring the fact that an ultrasound field develops side lobes which are not aligned with the scan direction, in most applications it could, however, result in image artefacts. Figure 1.7 displays a theoretical set up of the transducer and several objects spread over an opening angle of 60 degrees. Scanning the central 0-degree line, one could expect to receive an echo signal from screws one and two (placed on this central line) only. In case the two first order side lobes (yellow zone) generate enough intensity (i.e. pressure of sound wave is high enough) to reflect an echo, the transducer will additionally record echo signals from objects three, four and five. Since it is assumed that the signal represents one single line of the main beam to be reconstructed in the greyscale image, the angular spatial information of where the objects are actually located, is ambiguous and therefore lost [21]. All five objects would be depicted on the 0-degree line, falsely positioning the objects detected by a side lobe [21].



**Figure 1.7:** Exemplary Experimental Set Up and its theoretical Echo Signal Response

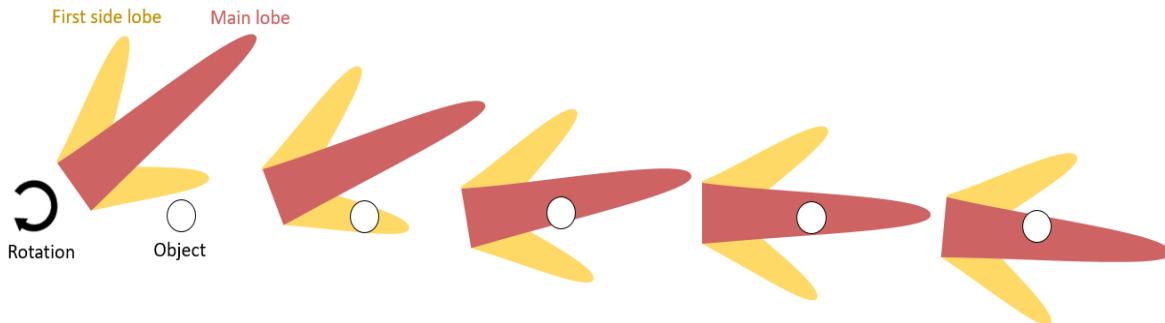
Furthermore, the maximal peak of the side lobe signal is proportional to its intensity, being significantly smaller than the main lobe signal's peak. The sum of the resulting single line signal, however, will then falsely be interpreted as showing objects of different materials (surface with different impedance values).

*Point Spread Function.* Regarding this problem from another angle, this means that the information of one single point source is spread over a slice in one an image direction orthogonally to the transducer's surface (along the beam axis) in the  $\varphi, r$ -plane represented by one line of pixels in the reconstructed 2D image as illustrated in Figure 1.8.



**Figure 1.8:** Transducer's Pressure Field, Intensity Profile along one Z-Position and its reconstructed Grey-Value Image Line when scanning a Point Source placed on the Z-Axis

As the transducer rotates from  $+ \varphi$  to  $0^\circ$  along the scan angle (see Figure 1.9), a fixed object is first captured by the first side lobe (neglecting side lobes of higher order here), then unregistered in the first zero of the intensity function and finally captured by the main lobe (the second half of the rotation acts accordingly).



**Figure 1.9:**  $(+\varphi, 0^\circ)$  Pressure Field of Rotating Single-Element Transducer (simplified)

The behaviour of how a point object is depicted in the reconstructed image is described by a Point-Spread Function (PSF). It defines how wide a true point is blurred by the main lobe and to which extent (intensity) it is registered by the side lobe(s). A resulting reconstructed image is therefore equal to a convolution of the true point-like object and its PSF. In the relevant literature for deconvolution models, the imaging process is described as a space-invariant linear system [23]:

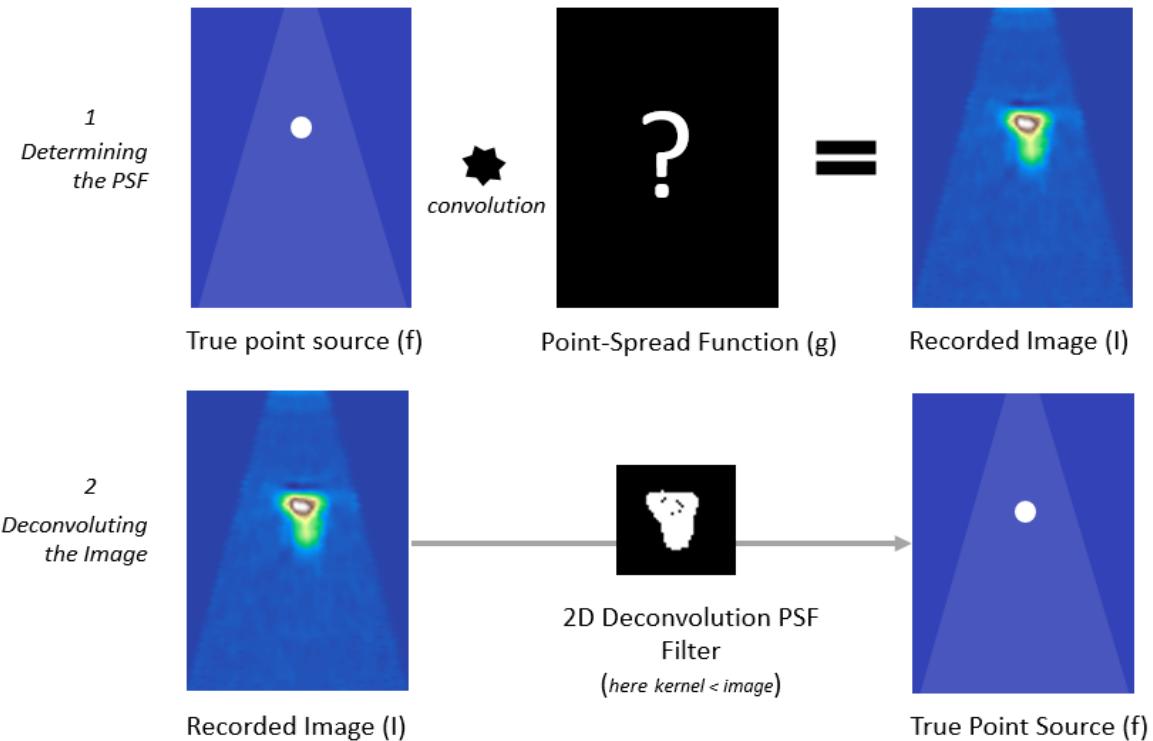
$$I(x, y) = f(x, y) * g(x, y) \quad (3)$$

Where  $I$  is the recorded image,  $f$  the object, “ $*$ ” the convolution operator and  $g$  the Point Spread Function.

Knowing a system’s PSF establishes the possibility to recursively reconstruct the true position of the point source by the inverse convolution (deconvolution). Applying deconvolution filters to an image in order to inverse the optical distortions and hence reduce the blurring is a vital step in medical imaging to enhance the image quality [23]. Figure 1.10 schematically illustrates the basic process of image deconvolution. Firstly the system’s PSF needs to be determined. Secondly, the computed PSF-filter is applied to recorded images in order to deblur them.

The PSF and image can be transposed to the frequency domain (Fast Fourier Transformation) which simplifies the convolution operator to a multiplication and its inverse deconvolution problem to a division [19]. Afterwards, it is retransformed into the time domain. This widely-applied approach was famously proposed by Norbert Wiener in 1949 [24] and was coined as the Wiener filter [23].

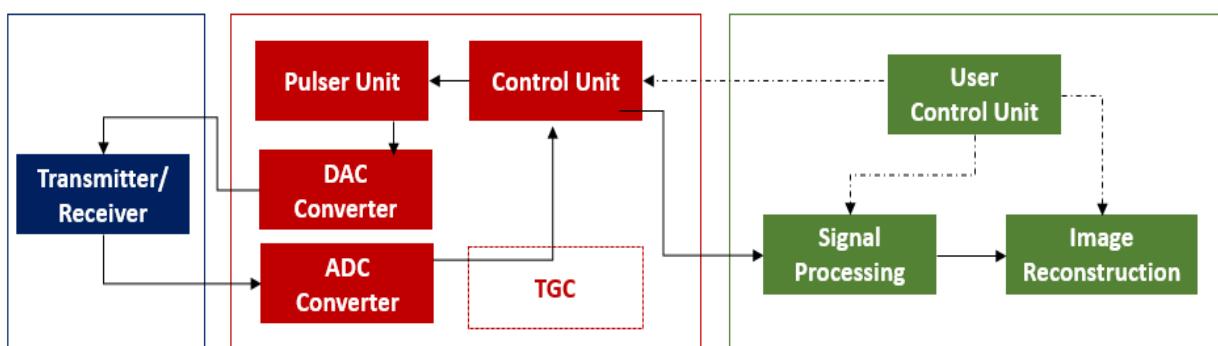
Deconvolution is widely applied in the literature by estimating the PSF from the image (the output of the system) not owning explicit knowledge of it. This thread of research is called *blind deconvolution*. Determining the PSF by directly measuring the scattering behaviour, approximating the “impulse response”, of a point source however is allotted to the field of *non-blind* deconvolution [23].



**Figure 1.10:** Schematic of Image Convolution and its Inverse Operation “Deconvolution” Process

### 1.3.3 Electronic Components of a Basic Ultrasound System

Which electronic elements does a simple ultrasound system consist of? There are several components necessary to build an ultrasound system which are depicted in the block diagramme in Figure 1.11. A pulser unit generates the necessary high voltage pulse applied to the piezo-element. Since the transducer emits and records analogue signals with a piezo-electric element, a digital to analogue converter (DAC) and an analogue to digital converter (ADC) are required. Time Gain compensation (TGC) components can be used to amplify the signal exponentially over time to counter losses due to dissipative effects. A control unit (e.g. an FPGA Chip), steers the pulser unit. The red block's function depicted in Figure 1.11 is realised with the Un0rick board in this project.



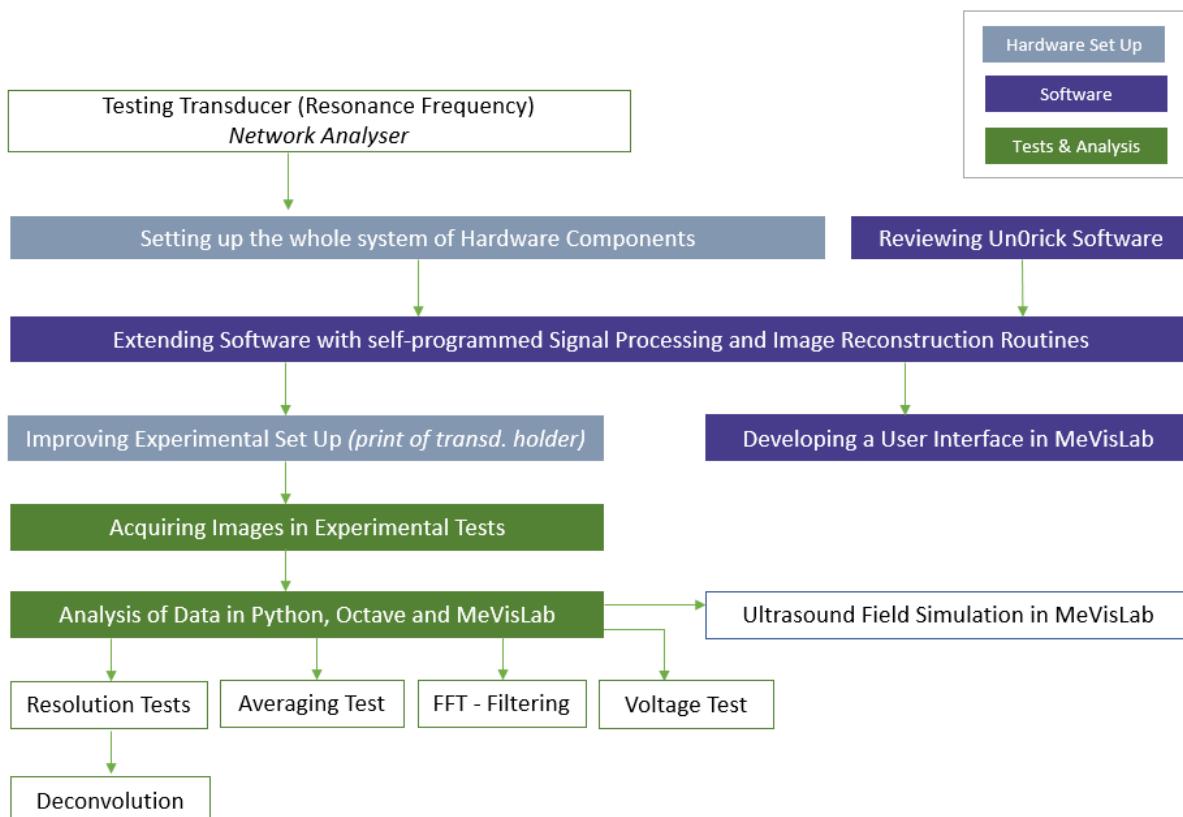
**Figure 1.11:** Block Diagramme of Basic Ultrasound System (adapted from [12])

Acquired digital raw signals require processing such as filtering before they can be used for the reconstruction of an ultrasound image. The overall ultrasound system is centrally controlled by a user control unit. These functions indicated in green in Figure 1.11 can be executed by the RP. In the following, the electronic hardware components used and signal processing routines developed for this project are introduced in more detail.

## 2. Materials and Methods

To implement and characterise such a proposed Low-Cost Ultrasound System as seen in the block diagramme (Figure 1.11) several complementary elements of development are required. This includes the materials and electronic hardware components and the software to control the system. Finally, the system requires testing and analysis to explore if and how it responds to parameter changes.

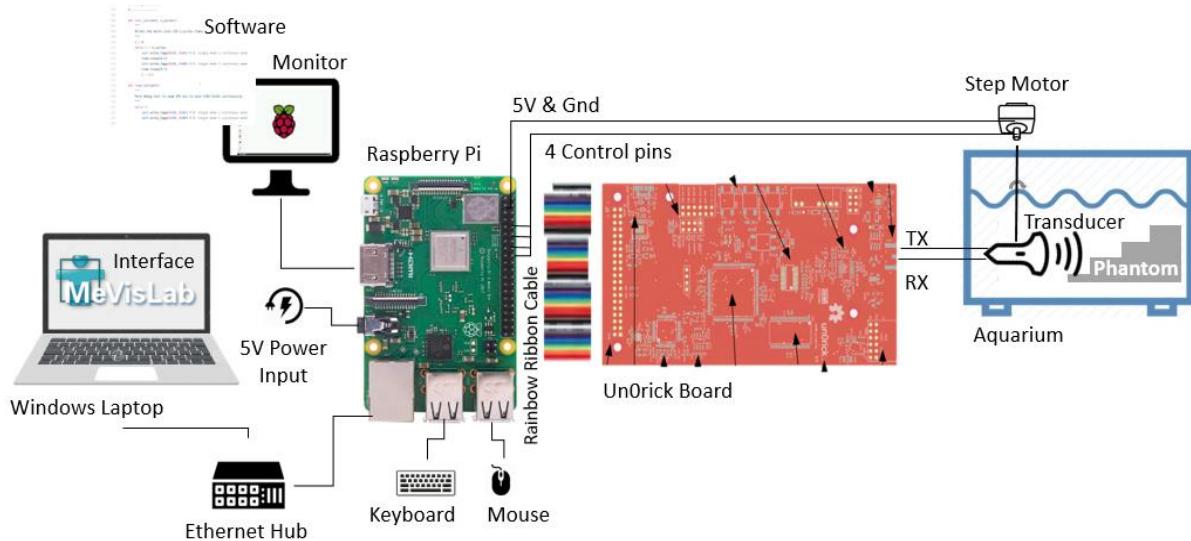
Figure 2.1 lays out the pursued methodology of characterising the LCUS.



**Figure 2.1: Methodology of Implementing the Low-Cost Ultrasound System (LCUS)**

## 2.1 Hardware Architecture of the Low-Cost Ultrasound System

Figure 2.2 below shows the hardware set up of the Low-Cost Ultrasound System shows all the vital components needed to set up the LCUS and how they are connected.



**Figure 2.2: Hardware Architecture of the Low-Cost Ultrasound System – Components & Connections**

### 2.1.1 Raspberry Pi

**Hardware and Interface Specifications.** A Raspberry Pi 3B+ Model (RP) is used [Raspberry Pi Foundation, UK, Cambridge]. The RP uses a 5V power input at 2.5A DC and provides 4 USB-2.0 inputs, 1 full-size HDMI input, 1 Ethernet-over-USB-2.0 and 40 GPIO pins for general use [25]. USB inputs were used for connecting the mouse and keyboard to the RP. The HDMI input is connected to a monitor. An extension pin board (not depicted in Figure 2.1) provides the possibility to use a 40-way rainbow ribbon cable [26] connecting the GPIOs and the UnOrick's 40 GPIO pins, whilst still be able to access the pins, which are unused by the UnOrick, to connect to the step motor. For controlling the “5V 28BYJ-48 ULN2003” step motor [MAKERFACTORY/ Conrad, Germany, Hirschau], 4 pins on the extension board are used (specified in **MewisUS.py** introduced in Chapter 2.2).

**Software.** The RP is operating on an Raspbian OS (Linux port) which is flashed onto a micro-SD-card. The newest Raspian Image [28] “Raspbian” Buster with desktop and recommended software, version July 2019] was downloaded from the official website. It controls the UnOrick ultrasound board as well as the rotating step motor. Signal and image processing can be optionally conducted on the RP or externally on a Laptop PC in the MevisLab Software (see Chapter 2.2).

### 2.1.2 Un0rick Board

Ultrasound acquisition and processing with open source hardware : un0rick.cc

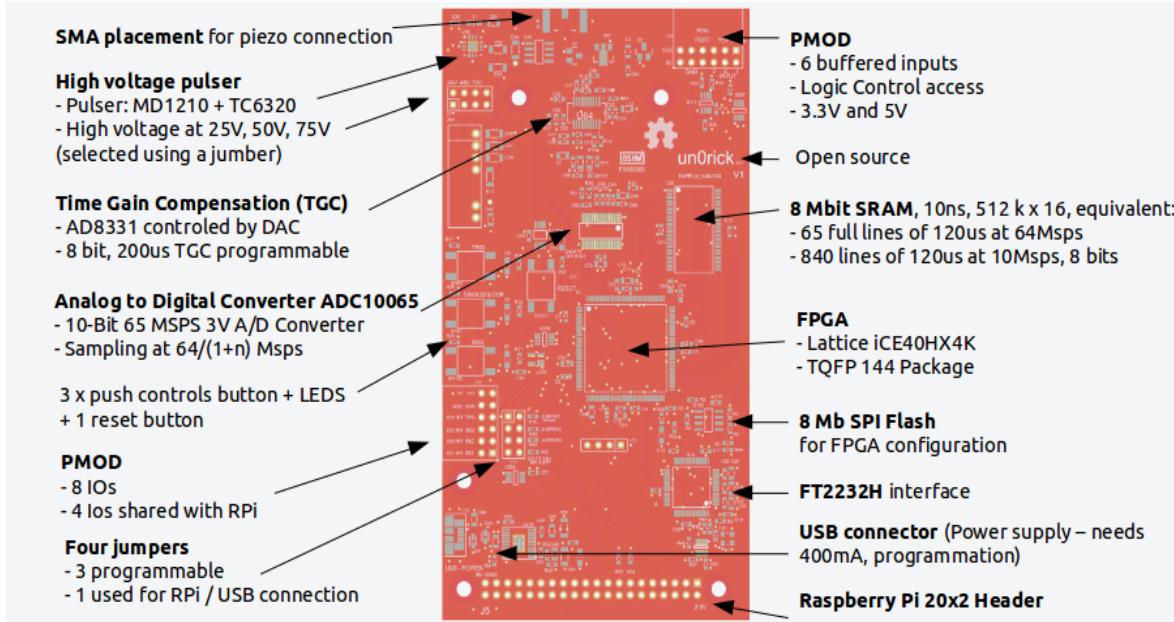
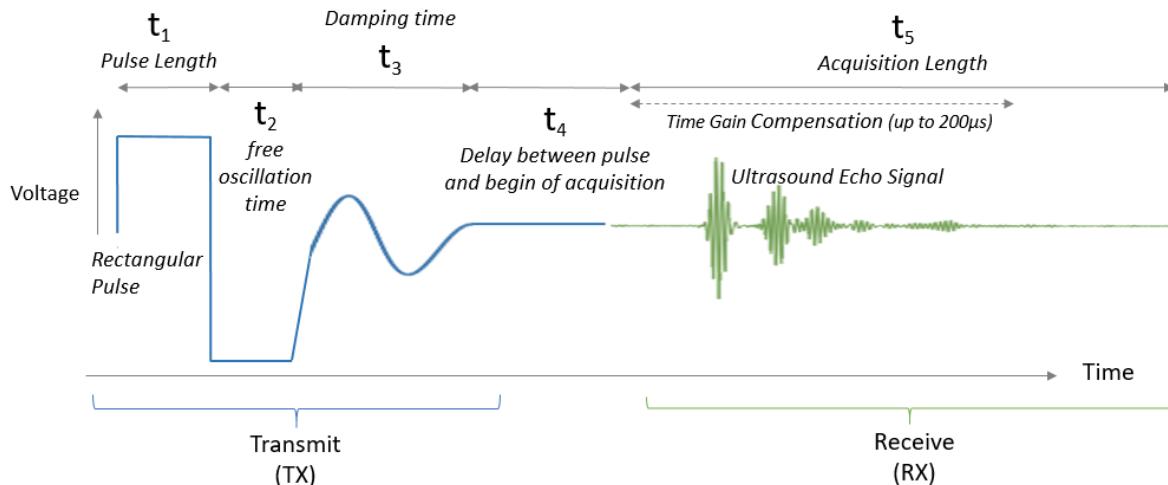


Figure 2.3: Un0rick Board's Interfaces [29]

**Un0rick Hardware and Interface Specifications.** The Board is controlled by a FPGA unit. It possesses a memory unit and an ultrasound programming unit including a VGA (controlled by DAC), a pulser unit and an ADC unit. Data is formatted over 2 bytes, with 10 bits per sample, 2 bits for line trackers, 4 bits of IOs and 2 bits for tracking. The un0rick board requires a 5V (at 350mA-450mA) input voltage provided either via USB or RP (SPI selection via jumper). The pulser voltage can be selected at 3 levels (24V, 48V and 72V) via a jumper. The FPGA and other logics on the board are operating at 3.3V [29]. The board version *Un0rickV1.1* utilised provides 2 separate SMA connectors, one for the transmission channel (TX) and one for the receive channel (RX). Since only a single-element transducer is used for experiments, the TX- and RX-Channels were effectively re-interconnected via SMA to BNC adaptors (T-pieces and cables), not taking advantage of the possibility to separate the channels. The pulse generated by the board is rectangular (voltage on /off).

**Un0rick Software and Ultrasound Acquisition Parameters.** The Un0rick project provides control software which can be downloaded from the project's website, offering processing routines for the acquired raw data, signal processing functions and data storage options. The open-source Un0rick software, more importantly, allows for programming the FPGA with the following ultrasound *pulse parameters* (partly depicted in Figure 2.4):



**Figure 2.4:** The Ultrasound Pulse generated by UnOrick Board and its Pulse Parameters

$t_1$  is equal to half of a wavelength. Pulsing at a frequency of 1 MHz translates into  $t_1 = 500$  ns. To allow one full oscillation,  $t_2$  is set to 500 ns in these experiments and  $t_3 = 1000$  ns to allow a full period for damping. The pulse parameter  $t_4$  is determined by the window one wishes to acquire. To avoid recording the fuzzy field behaviour near the transducer (see Chapter 3.3), it was set to 30.000 ns.  $T_5$  is limited by the experiment set up's maximal length in z direction. To cover around 30 cm,  $t_5$  was set to 320.000 ns. TGC is not applied in the tests.

The UnOrick board itself can be acquired in the project's online shop and software is openly accessible on a public GitHub repository [31]. The detailed documentation can be found on a GitBook page and on the project's webpage [29]. In a slack channel, Luc Jonveaux offers an online support environment. Throughout the project, he offered helpful advice on how to use the UnOrick board.

### 2.1.3 Transducer

A cylindrical (piston) ultrasound transducer of  $d = 30$  mm diameter was kindly provided by Prof. Christakis Damianou of the Cyprus University of Technology. It consists of a single piezo-electric element to generate (transmit) and record (receive) ultrasound signals. Its operating frequency is 1 MHz (see S11 Parameter Test, Chapter 3.1). The transducer is coated in Sonox 8 ceramic [MEDSONIC LTD, Limassol, Cyprus] and has an added epoxide resin vibration compensator.

### 2.1.4 Measuring Devices: Oscilloscope, Network Analyser and Signal Generator

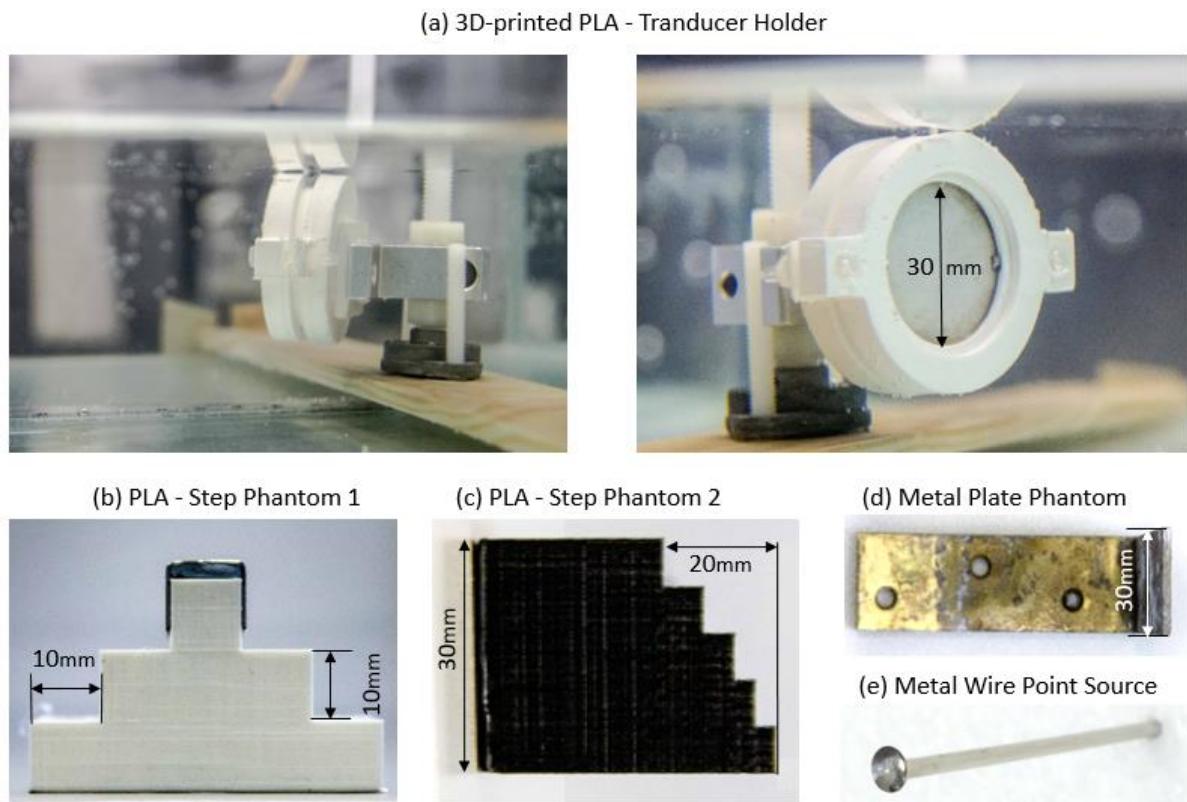
An oscilloscope [Tektronix Model Tektronix MDO 3034 Beaverton, US] is used to measure voltage signals throughout the project and testing the functionality of single components of the ultrasound set up such as the UnOrick board. To test the single element transducer for its functionality, a signal generator [Tektronix AFG 3022C Dual Channel, Beaverton, US] is used. Since the UnOrick Board is only capable of generating rectangular pulses, a rectangular one-cycle burst pulse is applied to the

transducer. The signal generator could only generate pulses of up to 5 Volts, hence the Un0rick board pulse conditions could not be simulated (the Un0rick board pulses at 24V, 48V or 72V).

In order to draw a more accurate picture of the transducer's resonance frequency band, the network analyser [Model NA Agilent Technologies E5061B (100 kHz-3 GHz), Santa Clara, US] is used to scan its frequency spectrum and measure the S11 parameter of the system. The results of this test are shown in Chapter 3.1.

### **2.1.5 Phantoms, Transducer Holder & 3D Printer**

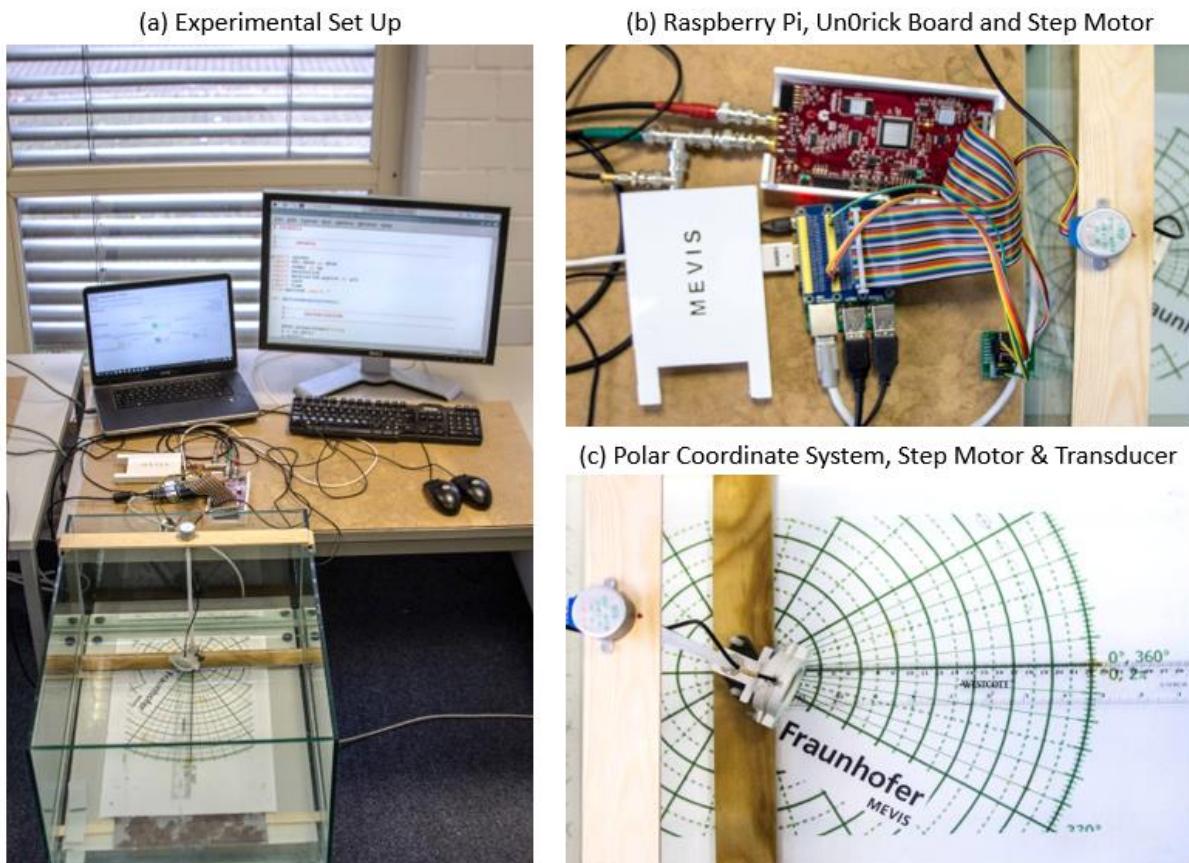
With respect to the investigated properties of the system, several different phantoms made of various material in diverse arrangements are used to acquire images. A readily available metal plate is used as a scan object (Figure 2.5 (d)). Other phantoms are 3D-printed with an Ultimaker-2 3D-printer [Ultimaker B.V., Netherlands, Utrecht] using mono-coloured PLA material (b) & (c). Being able to print a minimal layer thickness of 0.4 mm [32], small-structured step phantoms for transducer resolution tests could be printed with satisfying tolerance level. 3D-printing is also used for producing a holder which encapsulates the single-element transducer (a) and a case for the Un0rick board (Figure 2.5, "Mevis" inscription).



**Figure 2.5:** (a) 3D-printed Transducer Holder and (b)& (c) Step Phantoms as well as the (d) Metal Plate Phantom and (e) Metal Wire ( $d = 3 \text{ mm}$ ) Point Source used in Experiments

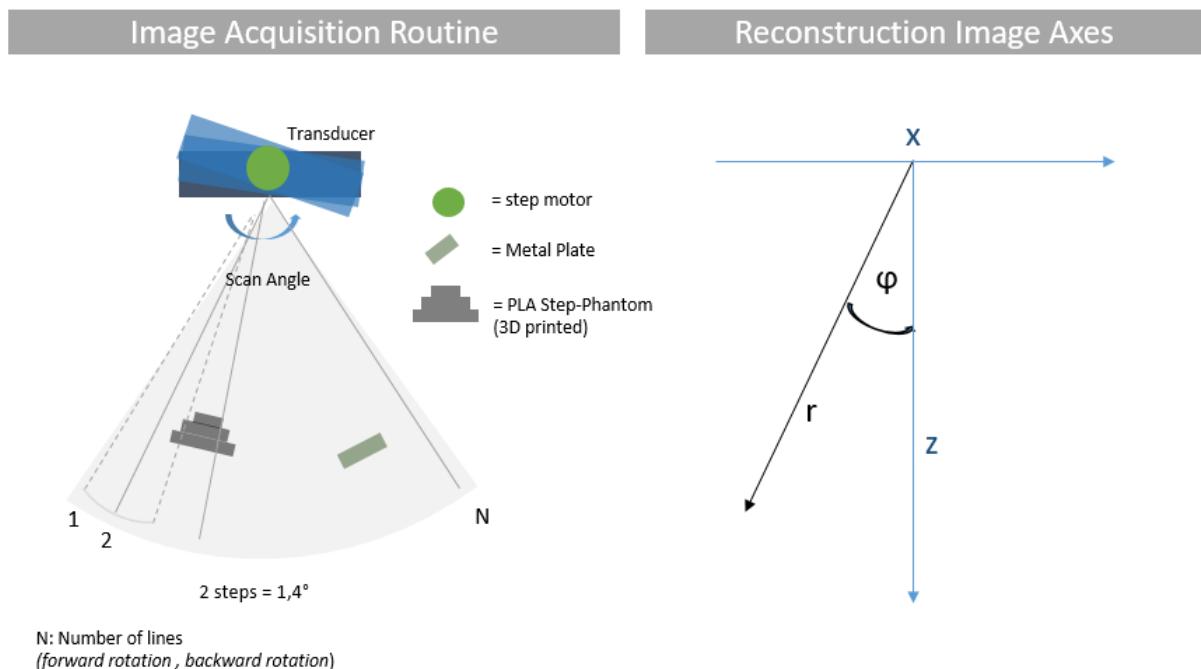
### 2.1.6 Experimental Set Up

The Ultimaker-2 is, moreover, used for rapid prototyping of the transducer holder (see Figure 2.5). This Holder was attached to the central spindle which is rotated by a simple and low-cost 5 V-step-motor [5V 28BYJ-48 ULN2003]. On its other end the spindle is mounted onto a swivel to ensure precise rotational steps. The step motor itself was held by a wooden construction glued together to fit onto the edges of a glass aquarium (dimensions: 30 cm x 40 cm x 55 mm). Remodelling human body conditions, experiments are conducted in an aquarium under water (including the waterproof transducer) at room temperature. Underneath this aquarium, a polar coordinate system (c) was placed on top of a metal plate so that magnets are prevented from floating, holding the phantoms to the ground of the aquarium.



**Figure 2.6:** (a) *Experimental Set Up*, (b) *RP, UnOrick Board and Step Motor*,  
 (c) *Polar Coordinate System, Step Motor and Transducer*

Figure 2.7 schematically explains the image acquisition set up of the Low-Cost Ultrasound System. As the transducer rotates along the scan angle (= 2 times opening angle), it acquires signal lines in a forward rotation and backward rotation shifted by one step.



**Figure 2.7: Image Acquisition Set Up and Image Axes as used for the Image Acquisition Routine**

### 2.1.7 Further Equipment

Further materials used in the set up are an HDMI-capable monitor and mouse and keyboard attached to the Raspberry Pi as seen in Figure 2.6. Jumper Cables and a pin extension board for the RP are needed to accommodate the connections between the RP to the step motor and the RP to UnOrick board. Since both, the oscilloscope as well as the network analyser, possess BNC connectors and cables, SMA to BNC adaptors are used. The transducer also requires a female to female SMA adaptor. For data transfer without Ethernet, a conventional USB flash drive is used.

### 2.1.8 Estimated Cost of Low-Cost Ultrasound System

How much does the Low-Cost Ultrasound System cost? Figure 2.8 lists the costs (the transducer is not bought, hence its potential price needed to be estimated) of all components of the ultrasound system. In total, 537,50 EUR are spent on the hardware components.

Component	Price
UnOrick Board	440 €
Raspberry Pi	35 €
Single-Element Transducer	50 € <i>(estimated if produced industrially)</i>
Step Motor	2.5 €
Materials experim. set up	10 €
Software (MeVisLab, UnOrick, Octave)	free
<b>Total</b>	<b>537,50 €</b>

**Figure 2.8: Costs of Low-Cost Ultrasound System**

## **2.2 Software Architecture of the Low-Cost Ultrasound System**

The illustrated hardware is complemented by software. The pulse generation of the UnOrick board, the image acquisition and the step motor rotation requires controlling. Furthermore, functions such as sending and saving acquired data, signal and 2D image procession are also executed by software suited to the system.

### **2.2.1 Python**

Python is a widely used programming language in scientific research [33]. It provides an easy syntax and a multitude of open-source extension packages for various mathematical, visual and other applications. For this project, all code is written in Python 3. The UnOrick's software was developed in Python 2. It required conversion to Python 3 since, as of January 2020, the Python Foundation will cease support for Python 2 [33]. Adapting the code to Python 3 additionally ensures compatibility amongst all components of the systems. **Numpy**, **SciPy**, **Matplotlib**, **Imfit.Model** and other Python packages used in this master's thesis are free open-source software [34,35].

### **2.2.2 Octave**

GNU Octave is a free and open-source scientific programming language for numerical computation. It provides built-in plotting and visualisation tools and - very similar to Python - packages for signal and image procession [A]. The **image** and **signal** packages are used for deconvolution of images in the GNU Octave-5.1.0.0 (GUI) version.

### **2.2.3 MeVisLab**

The MeVisLab suite is a powerful environment for modular software development image processing research, especially focussing on medical imaging [36]. It allows for fast integration and testing of new algorithms and development of fast prototypes [36]. MeVisLab makes use of several well-known third-party libraries and technologies, relevant for the purpose of this master's thesis, the scripting language Python [36]. It provides the opportunity to programme macro-modules which can embed Python scripts. Several macro-modules can be graphically connected on a canvas to form a MeVisLab network. Networks in turn can be grouped into macro-modules. For every macro-module, a graphical user interface (GUI) to control the parameters of the module allows to specify input, output or button fields to trigger commands of its respective Python script. The software is used to provide a GUI for the Low-Cost Ultrasound System. A macro-module called *LowCostUS* is created containing three sub-modules. The *MevisUSParameters* Module (see Chapter 2.4) allows the user to specify acquisition parameters. The *SendReceive* module establishes a web socket connection, sending and receiving data from and to the Raspberry Pi.

## 2.2.4 Software Development

Three major components of software development can be distinguished.

- Software from the UnOrick open-source project is used to operate the Low-Cost Ultrasound device. The documentation of the software is detailed but not complete. Hence, it is important to review the structure and methods provided, since the UnOrick project is still ongoing, constantly updating and improving its own software.
- Missing functionalities are programmed. This includes most of the signal processing routines and image reconstruction methods. Integrating the self-programmed methods into the UnOrick software allows to reconstruct 2D ultrasound images.
- To ensure the usability of the Low-Cost Ultrasound System, the routines are implemented in the MeVisLab module **LowCostUS** to provide a graphical user interface.

The implemented software architecture of the ultrasound system is illustrated in Figure 2.9. Steps 1 to 7 depict the interaction of all scripts and their respective function. In the following paragraphs, the three main software development components leading up to this integrated architecture are explained in more detail.

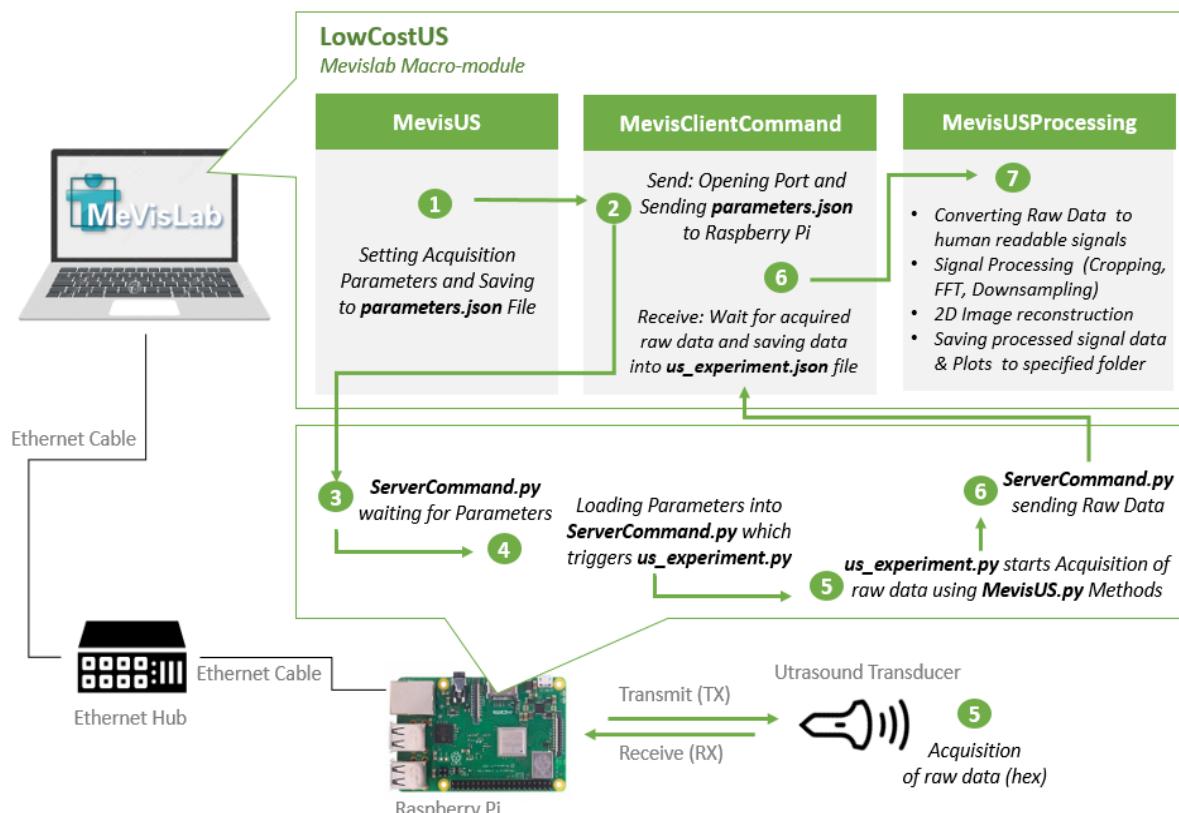
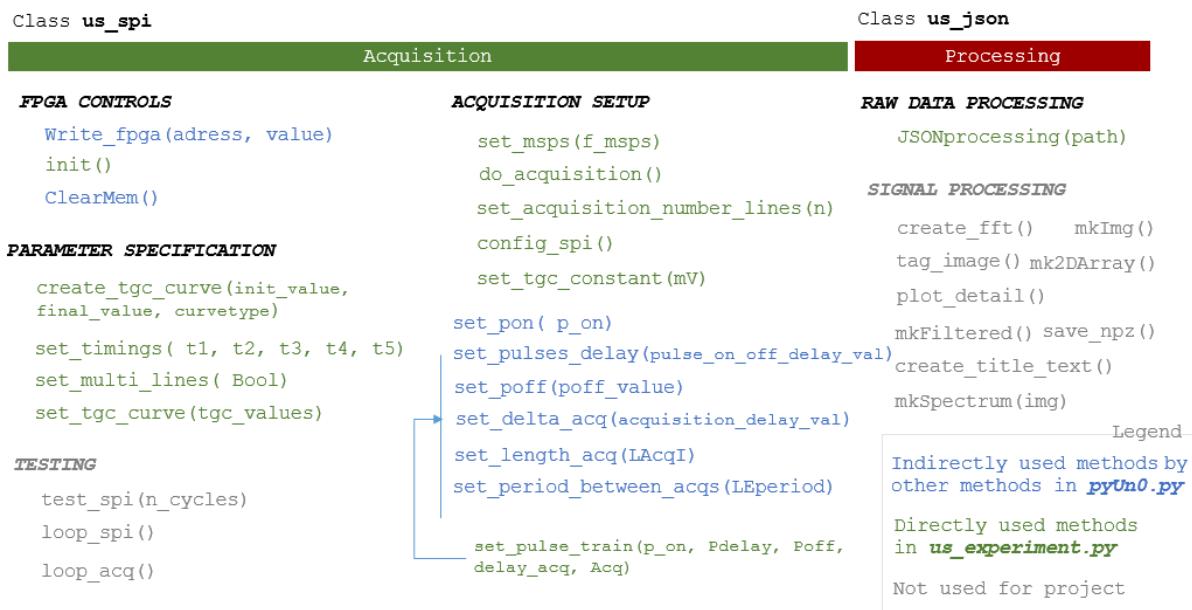


Figure 2.9: Software Architecture: How to acquire an Ultrasound Image with the UnOrick Board

#### 2.2.4.1 UnOrick-Software Review

Downloaded from the project website, the **`pyUn0.py`** script is the heart of the acquisition software for the UnOrick board, providing all necessary methods for acquiring and processing ultrasound signals. The software entails a `class =us_spi` which includes a few methods communicating with the FPGA via the SPI. Commands such as `write_fpga()` and `ClearMem()` are writing into the FPGA's registers. It also provides methods for setting ultrasound acquisition parameters (see Figure 2.10) and UnOrick board testing functions\*. The second class of the unOrick's project software `class =us_json()` offers functions such as `JSONprocessing()`. Apart from this method, none of the methods in this class are used (grey colour-coded in Figure 2.10 are not used). Figure 2.10 shows an overview of the available methods in the script and indicates which methods are selected for implementation of the LCUS.



**Figure 2.10: Reviewing and Selecting the Functions of the UnOrick-Software**

\* The use of these functions is explained in Figure 2.11, Paragraph „Acquiring an ultrasound image“.

#### 2.2.4.2 Extension of Software with Signal Processing and Image Reconstruction Routines

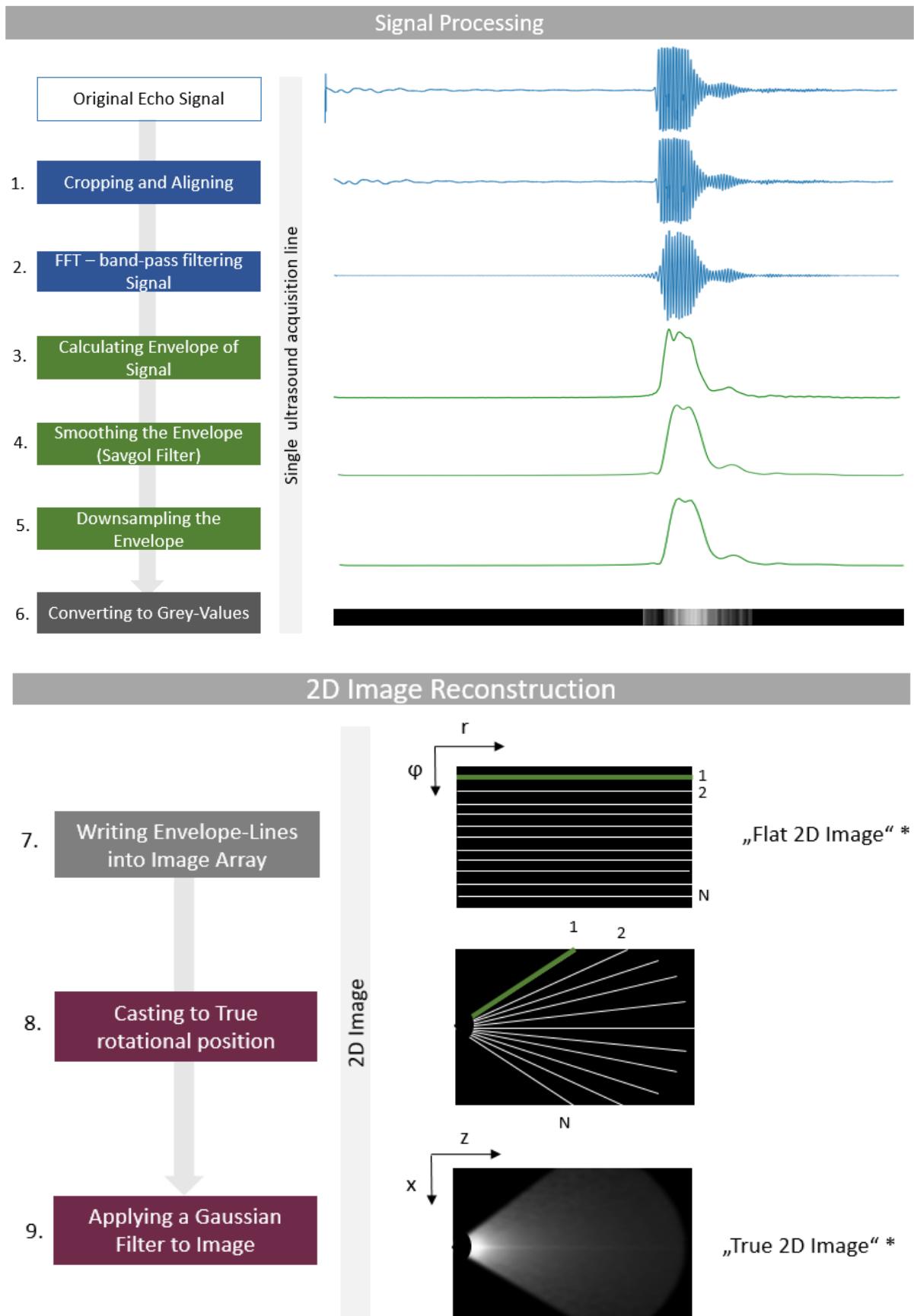
Having selected the UnOrick methods to be used for the implementation of the LCUS, missing routines necessary for signal processing and image reconstruction are developed. They are merged with the UnOrick-software into a method collection script called “**MevisUS.py**” (see Figure 2.9, step 5). The implemented signal and image procession methods are depicted in Figure 2.11, where the result of each step is shown by means of a signal with one echo peak.

1. Firstly, depending on the length of the acquired signal (pulse parameter:  $t_5$ ), the signal requires cropping on both ends. Only representing the region of interest, ensures recording a minimal amount of data points. It improves the speed of consecutive image reconstruction. Additionally, as the original signal in Figure 2.11 shows, the system acquires the tails of the pulse send through the transmission channel. Since in this project the receive and transmission channels are inter-linked, cropping is one way of eliminating the unwanted pulse signal at the beginning.
2. Secondly, a FFT-band-pass filter is applied to decrease the level of noise in the signal. Systemic oscillations are also eliminated with this step.
3. In order to be able to represent ultrasound signals as grey values, an envelope of the positive part of the signal is then computed. This is done by determination of the signal’s *Hilbert Transform*.
4. Subsequently, smoothing the envelope function eliminates outliers. For the smoothing, a *Savitzky-Golay*-filter is implemented.
5. The signal still consists of a large amount of data points (each signal array approx. 90.000 in the experimental tests), slowing down image reconstruction. Hence, down-sampling is undertaken.
6. The down-sampled and smoothed envelope function of the signal can now be interpreted as an image array of grey values where the smallest value is allotted to 0 and the peak value of the reconstructed image to 1 (depicted as grey-value converted line in Figure 2.11). The signal is normalised to the range [0,1].

In order to reconstruct a two-dimensional grey-value ultrasound image from the processed signals, the individual envelope functions of each line are written into a 2D image array and the transducer is rotated between the acquisition of any two subsequent lines.

7. Each signal is written into one row of an image matrix first, not considering the rotational set up to be able to preview the data and supervise the reconstruction. This step was introduced because the reconstruction of this 2D image is helpful for debugging and offers an alternative for representing the data\*. These image arrays are onwards titles as “Flat 2D Image”.
8. In a second step, the signals are then cast to their actual spatial position in the image according to the rotational experimental set up. When mentioning this image representation, it is referenced to as “True 2D Image” in the following tests.
9. Finally, a low magnitude Gauss-filter (default values:  $\text{kernelX}=\text{kernelZ}=1$ ) in x and z direction is applied to the image.

\*This representation of image data is used in Chapters 3.3., 3.4 and 3.7.



**Figure 2.11: Implemented Signal Processing and Image Reconstruction Process**

\* The reconstruction algorithm can be found in the “Annexe” (Chapter 7)

For the FFT-filtering, the **numpy** Python package [35] is used. The calculation of the Hilbert transform as well as the Savitzky-Golay smoother and the down-sampling in Python are implemented by using the **SciPy.Signal** Python package [34] which provides functions and methods for signal processing. For generation of the image plots, the LCUS used **matplotlib**. The command `matplotlib.imshow()` automatically interprets the values as normalised values between 0 and 1 (grey-values or according to other specified colour map) and allows to apply the Gaussian-filter. Contrarily to the dominant convention of representation of image arrays in grey-values, the ultrasound images rendered by the LCUS are presented in colour maps “terrain” (predominantly used in geological maps since it has the largest colour spectrum) and “viridis” (default Python colour map)

#### *Acquiring an Ultrasound Image*

To acquire an image (Figure 2.9 – “Software architecture”, steps 4 and 5) with the experimental hardware set up depicted in Figure 2.6 and 2.7 in Chapter 2.1 and using afore-mentioned signal processing methods, one needs to execute the **us\_experiment.py** python script. Its code is displayed in Figure 2.12.

- Firstly, the FPGA Chip on the UnOrick Board requires initialisation and the default values for parameters are configured.
- Secondly, acquisition parameters are specified (Step 1 in Figure 2.9). This concerns pulse parameters ( $t_1-t_5$ ), UnOrick board parameters (voltage, sampling frequency, TGC, multiline or single line acquisition) and experiment parameters (number of lines, scan angle).
- Thirdly, the FPGA settings are updated with the new parameters and the acquisition is started (steps 2-5 in Figure 2.9). During an acquisition loop, a single line is sampled (received), the data read out from the UnOrick-Board and saved into a `.json` object and the motor then rotated by 2 steps. The acquisition of one image is divided into two loops, one forward rotation (even lines) and one backward rotation (odd lines). In between the motor is rotated 5 steps forward and 6 steps backwards without line acquisition. This feature is required since the step motor cannot simply move forward one step (mechanically not feasible).
- After the acquisition, hexa-decimal data signals stored in the `.json` object are saved to a **data.json** file in the same folder.

With the software development components one and two, the software is in a self-sufficient state to acquire ultrasound images within the physical, digital and computational limitations of the system. Running the **us\_experiment.py**, it takes about 90 seconds until the reconstructed image is available.

```

Us_experiment.py

import spidev
import RPi.GPIO as GPIO
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import json
import time
from mevisUS import *    ← Import mevisUS.py methods

GPIO.setwarnings(False)
x = us_spi()
x.init()
x.fpga_initial_config() ← default parameters from config()

#path_parameters =
filename_parameters = str("parameters.json")          PARAMETER SPECIFICATION I

with open(filename_parameters, 'r') as json_file:
    data = json.load(json_file)

Experiment_Name= data["ExperimentName"]
tgcValue= int(data["acqParameters"]["tgcValue"])
tgcCurve= int(data["acqParameters"]["tgcBool"])
tgcCurveI= int(data["acqParameters"]["tgcCurveI"])
tgcCurveF= int(data["acqParameters"]["tgcCurveF"])

# ACQUISITION PARAMETERS

#timings
t1 = int(data["pulseTrain"]["t1"])# pulse duration
t2 = int(data["pulseTrain"]["t2"]) # delay time until damping
t3 = int(data["pulseTrain"]["t3"]) # damping time
t4 = int(data["pulseTrain"]["t4"]) # delay time until acquisition starts
t5 = int(data["pulseTrain"]["t5"]) # acquisition length

# other parameters
V_pp = int(data["acqParameters"]["Vpp"]) # 24V, 48V, 72V

# motor rotation
number_rot = int(data["acqRoutine"]["rotNumber"])
# number of rotations
steps_rot = int( data["acqRoutine"]["rotSteps"])
# motor steps per rotation, 1 step = 0.704 degrees

IMPORTS
INITIALISATION
PARAMETER SPECIFICATION I

```

```

# __ lines and sampling Parameters _____          PARAMETER SPECIFICATION II

multilines = 0 # 0: one line, 1: more lines
f = int(data["acqParameters"]["f"]) # frequency of ADC sampling;
n = 1 # number of lines

x.set_tgc_constant(tgcValue) # tgc constant
x.create_tgc_curve(tgcCurveI, tgcCurveF, tgcCurve) # tgc curve
x.set_multi_lines(multilines) # multi lines
x.set_mspis(f) # ADC sampling frequency
x.set_acquisition_number_lines(n) # number of lines
x.set_timings(t1,t2,t3,t4,t5) # set the pulse train
x.updateNacq # to update the number of data points to acquire.          FPGA SET UP

ACQUISITION AND ROTATION

# acquire the signal, readout the data, rotate and repeat
for data_reg in range(0,2*number_rot,2):
    #print("Line # " + str(data_reg))
    x.do_acquisition() # start the acquisition
    x.do_readout(data_reg)
    # reads the data from the board
    x.rotate(steps_rot,-1)
    # rotate(steps, direction), direction: +1 or -1 for ccw or cw rotation

    # rotate 1 step back
    x.rotate(5,-1)
    x.rotate(6,1)

    # acquire the signal, readout the data, rotate and repeat
    for data_reg in range(2*number_rot-1,-1,-2):
        #print("Line # " + str(data_reg))
        x.do_acquisition() # start the acquisition
        x.do_readout(data_reg) # reads the data from the board
        x.rotate(steps_rot,1) # rotate(steps, direction), direction: +1 or -1 for cc

    #rotate back
    x.rotate(5,1)
    x.rotate(6,-1)

path = "us_experiment"
x.do_export(path) ← Saving signal to json file          PROCESSING HEX DATA to RAW DATA

```

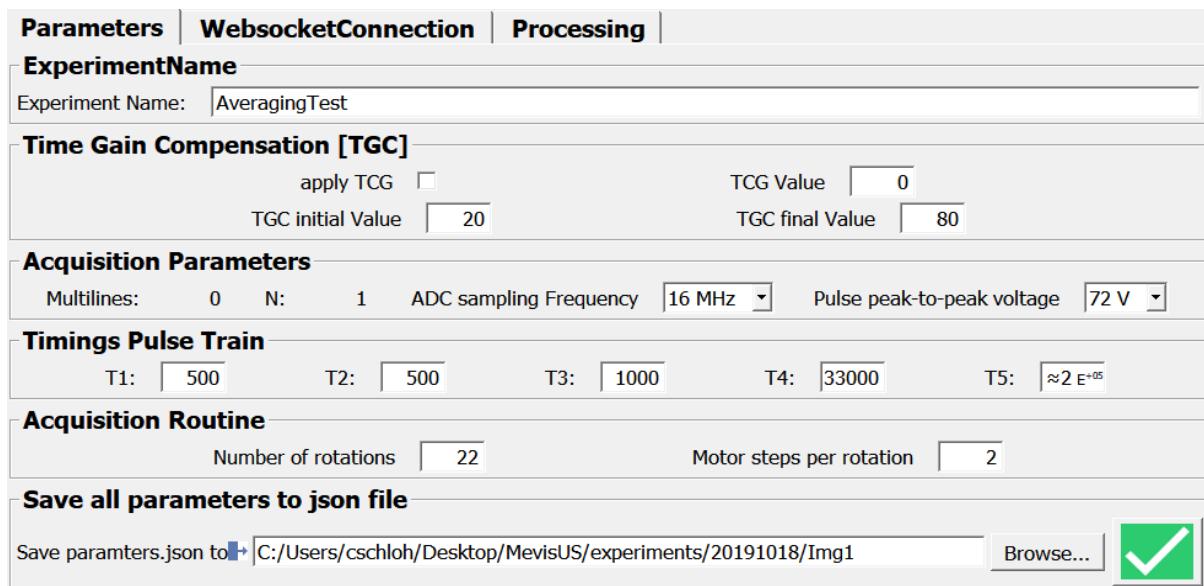
**Figure 2.11:** *Us\_experiment.py – Acquiring an Image with the Low-Cost Ultrasound System*

### 2.2.4.3 The MeVisLab Macro-Module “LowCostUS”

The MevisLab interface leads the user through the process of acquiring an image by limiting the selectable range of individual parameters (sliders, drop-down boxes, check boxes). It adds informative descriptions to the input fields by showing status updates and divides the imaging process into the three logically sound units: *parameters specification*, *acquisition* and *signal processing/reconstruction*. Each is represented by a macro-module (MevisUS, SendReceiveUS, MevisUSProcessing) and is grouped to the Macro-Module **LowCostUS**. The Interface window of the **LowCostUS** incorporates the corresponding three tabs and links the three modules as can be seen in Figure 2.9, steps 1,2,6 and 7. It links the so far described Python routines with a GUI and connects the Raspberry Pi and the Windows10 laptop by Ethernet, completing the software architecture described in Figure 2.9.

*Tab 1: Parameters*

Acquisition parameter information for conducting an experiment is entered. Firstly, the experiment name and TGC settings are specified. One needs to select the sampling frequency and pulse's peak voltage, parameters such as multi-lines are pre-set but nevertheless shown in the interface.

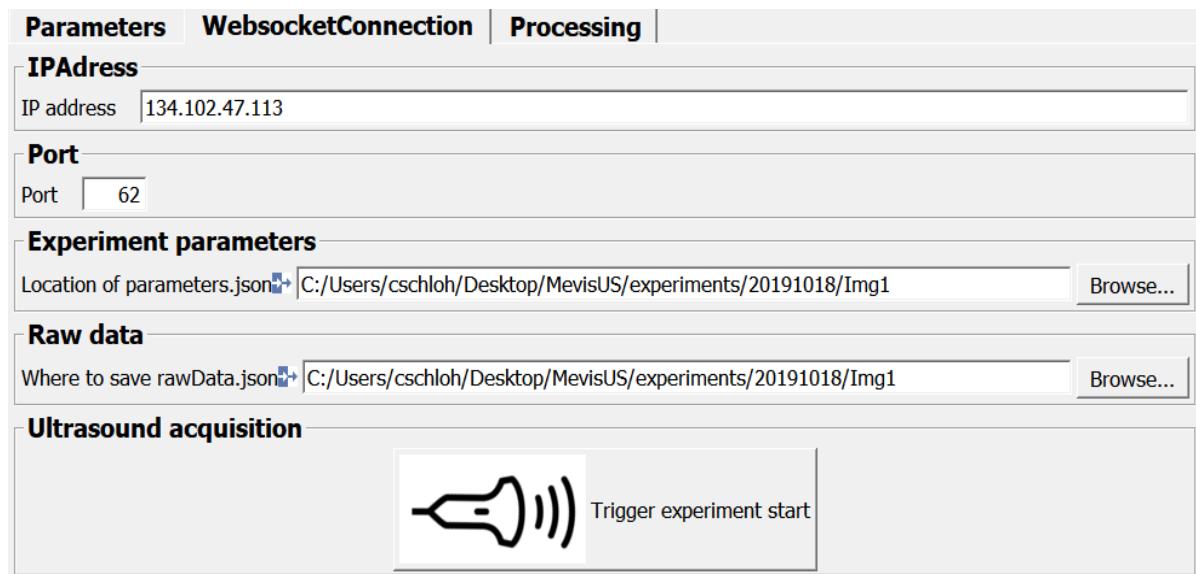


**Figure 2.12:** Tab1: Acquisition Parameters Interface

Finally, setting the pulse train parameters and the specifications of the acquisition routine, the parameters are saved to a **parameters.json** file to the entered file path by clicking on the green check button at the bottom.

**Tab 2: SendReceive Web Socket connection**

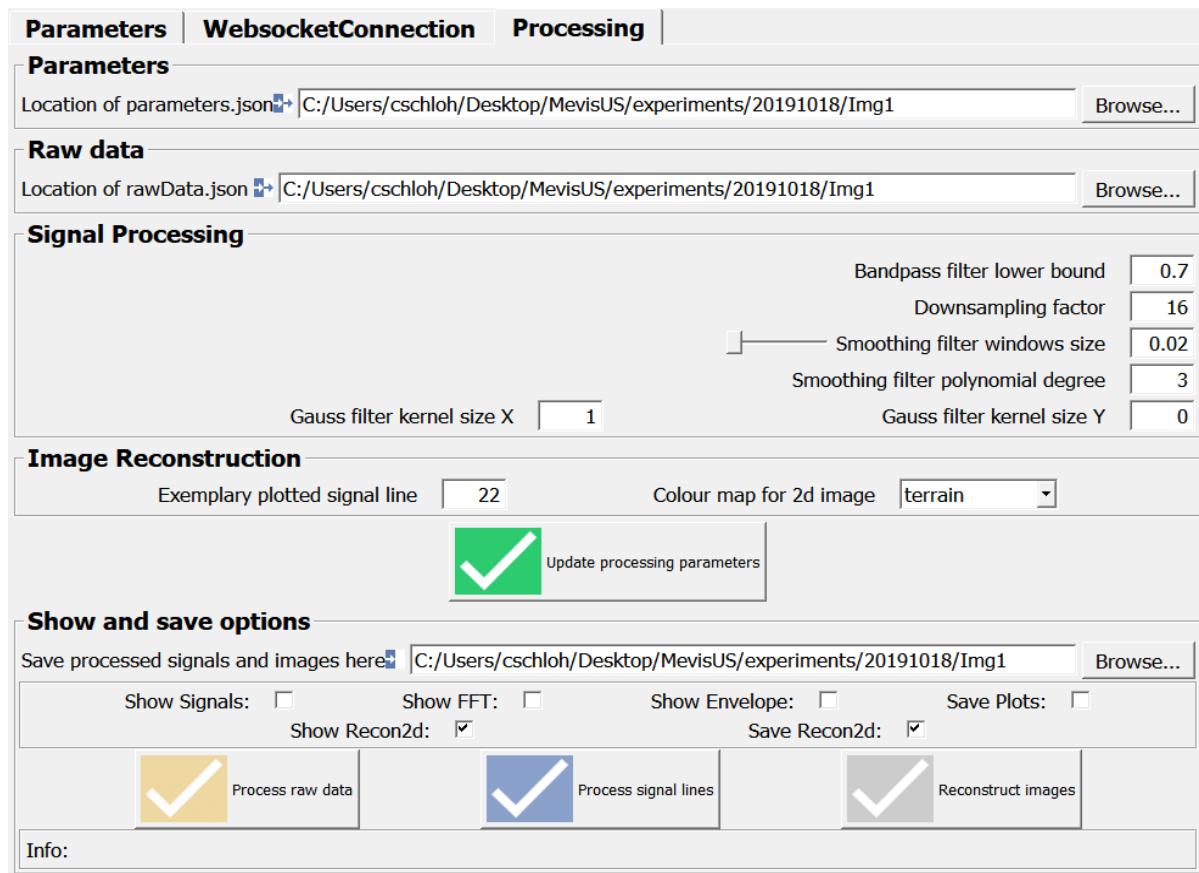
In order to establish the Ethernet connection between the Windows10 Laptop and the Raspberry Pi, the IP-Address of the Raspberry Pi and the number of the port used to transfer the data need to be entered to the fields. Both inputs are required to correspond to the information given in the “listening” script on the Pi. This is a mirrored version of the web socket module acting as server, waiting for the client module (SendReceive Web Socket macro-module in MevisLab) to send the **parameters.json** data file. By pressing the trigger “Experiment Start” button, the web socket connection is established, and the parameters are sent to the waiting server. Having received the parameters, the server script then triggers the actual acquisition (**us\_experiment.py** script., described above) which in turn loads the received acquisition parameters and loads and uses the methods specified in the **MevisUS.py** Script. After having done the ultrasound acquisition of N lines (number of acquisition lines), these are saved into a **data.json** which is send back to the now waiting client **SendReceive** MeVisLab module. The signals in the data file are hexa-decimal figures. Therefore, the first step in the next tab is the conversion to decimal numbers.



**Figure 2.13:** Tab 2: Web-Socket Connection Interface

### Tab 3: Processing

Due to the linked fields in the first and second module to the third module, the file path is automatically adopted into the field “filepath”. Signal processing and image reconstruction parameters can be updated by entering the information and triggering the grey button. If not done so, the programme uses default values. After having updated the parameters (which are also saved to the already existing **parameters.json** file), the user needs to click on the green “Process Raw Data Button” (see Figure 2.14). To further process the raw data signals, it can be selected which plots the user would like to pop up and which ones to save to the folder. Per default, the reconstructed 2D image is saved, the sub step plots are not being saved. After receiving a status update in the info box that the signals are processed, one clicks on the third (grey) button “Reconstruct Images” to receive the final ultrasound image. Figure 2.14 shows the graphical user interface.



**Figure 2.14:** Tab 3: Signal Processing and Image Reconstruction Interface

## 2.3 Methods of Experimental Testing

Physical and electro-technical properties of an ultrasound system are determined to measure the performance of the LCUS. According to the International Electrotechnical Commission (IEC) ultrasound working group, the determination of its resolution is an important aspect of characterising an ultrasound system [37]. The Point Spread Function (as described in the introduction Chapter 1.2.2) method is referred to in IEC standards [38], where PSF measurement is specified as the “characteristic response of the imaging system to a high contrast point target” (impulse response of the system). Other international initiatives for quality assessment of ultrasound systems (such as the Bavarian initiative “Sonobaby” and the “QA-Group” EFSUMB) propose testing the Signal-to-Noise (SNR) level of a system [38]. Thus the voltage, FFT-filtering and averaging tests are evaluated by determining the Peak-Signal-to-Noise\* ratio (PSNR). The IEC furthermore advises to test components such as the transducer individually from the system [37].

### *Initial Pre-Test*

In order to test if the single-element transducer functions, the signal generator - producing a sinusoidal signal - and oscilloscope are connected to the transducer. As a signal can be recorded, the transducer works. The UnOrick is tested by measuring its output voltage with the oscilloscope when operated. The results of the test confirmed that the UnOrick board generates voltages the characteristic pulse signal explained in Figure 2.4. Having assured these individual components are working, further tests are conducted.

### *Determination of the Transducer’s Resonance Frequency*

In electronics, for a one-port power circuit, the scatter S-11 parameter  $\Gamma$  describes the ratio of reflected power [39]. To determine the frequency at which the single-element transducer should be operated (resonance frequency of the transducer), the S-11 is measured. During the test, the Network Analyser sends power  $P_{in}$  into the piezo-element and then measures the intensity  $P_{reflected}$ . If  $\Gamma \neq 0$  over a range of frequencies, reflected in the transducer.

For this test, a frequency window with a span of 4 MHz and 2.2 MHz as central frequency (range 0.2 to 4.2 MHz) is scanned. The piezo-element is tested immersed in water. At frequencies with a maximum, the highest ratio of signal is reflected. This is the resonance frequency of the transducer at which it should be operated to perform optimally.

---

\* The PSNR here is defined as the ratio of the peak value of an Echo Signal in relation to the average Noise level. It is used for graphical evaluation of the results in Chapters 3.3 and 3.5.

### *Simulation of Transducer's Ultrasound Field*

The field of a simplified and ideal ultrasound beam of a piston transducer is described in the introductory Chapter. To investigate the theoretical behaviour of the single-element transducer's ultrasound field, the parameters available ( $d = 30\text{mm}$ ,  $f_{res} = 1\text{MHz}$ ) are entered into a *simulation tool for ultrasound beams* in MeVisLab. The tool simulates the ultrasound field with the parameters  $d = 30\text{mm}$  and  $f = 1\text{MHz}$  in water. It renders 2D plots of the axial and lateral behaviour of the transducers ultrasound beam. To explore the side lobe behaviour (lateral) and the focus point (axial) of the beam, the simulation results are shown in Chapter 3.2.

### *Analysis of UnOrick Board's Sampling Frequency and Voltage*

The voltage applied to the piezo-electric single-element transducer defines the height of the pulse's amplitude. A higher voltage results in pressure waves of higher amplitude emitted into the medium. This means that echo signals carry a higher intensity and the received signal's Peak-to-Noise-Ratio (PSNR) is potentially higher than pulsing at lower voltages. The UnOrick board allows the selection of 24 V, 48 V and 72 V.

In order to test if using the UnOrick board at higher voltages indeed improves the PSNR, the experimental set up depicted in Figure 2.6 (Chapter "Hardware Architecture", 2.1), where the PLA-printed step-phantom 1 (b) was scanned (step heights and depths: 10 mm) and a 3 mm thick metal plate (d), is used and scanned with fixed acquisition parameters, only varying the sampling frequency (16MHz and 64 MHz) and the voltages (24 V and 72 V). The acquisition parameters are pulse timings ( $t_1=t_2=500\text{ns}$ ,  $t_3=1.000\text{cns}$ ,  $t_4=70.000\text{ ns}$ ,  $t_5=250.000\text{ ns}$ ), the total number of lines is 45, the scan angle is 63 degrees, the band-pass filter ( $f_0 = 0.8\text{ MHz}$ ,  $f_1 = 2*f_0$ ) and the downsizing factor equal to the sampling frequency (16 or 64 respectively). No smoothing was applied to the signal to avoid misrepresentation of the data. The images are reconstructed with the standard reconstruction parameters as described in Chapter 2.1.2.

### *Signal Fast Fourier Transformation Frequency Spectrum Analysis*

For noise reduction outside frequencies of interest, a band-pass filter is typically applied in signal processing [40]. Another function of the band-pass filtering in ultrasound is to select the imaging mode of using the fundamental frequency (conventional imaging) or the second harmonic or higher (harmonic imaging) [40].

Frequency domain filtering methods process an acquired image in the Fourier domain to emphasise or de-emphasise specified frequency components. Low frequency components of a signal are depicted in an ultrasound image as blurred structures and high frequency components as sharp details and edges [41]. High frequency components carry the noise information.

In order to analyse the ultrasound device's frequency spectrum, a single line is acquired using the pulse timings parameters described in 2.1.3 at a Voltage of 72 V and sampling rate of 16MHz. No cropping, down-sampling and smoothing is applied. The signal acquired shows two unclear echo signals from two steps of the phantom 1 (b) used in the voltage test. For the calculation of the Fast Fourier Transform, `numpy.fft.fft()` and `numpy.fft.ifft()` Python methods are used.

#### *Application of Image Averaging*

An acquired image  $I(x,z)$  can be assumed as a representation of the ideal image  $F(x,z)$  of the scanned object and an added undesired component of noise behaviour  $N(x,z)$  [41]. Noise behaviour is assumed to be uncorrelated and random (white noise).

$$F(x,z) = I(x,z) + N(x,z) \quad (4)$$

If properly registered in stationary applications, sequence images can be averaged to reduce the noise [30] which leads to smoothing effects. The average image can be obtained as:

$$Fav = \frac{1}{K} * \sum_1^k F(x,z) \quad (5)$$

Where  $F_k(x,z)$ ,  $k=1,2,\dots,K$  represents the sequence of images to be averaged. With an increase in the number of images  $K$ , the expected value of noise reduction per pixel is factor  $\sqrt{K}$  [41].

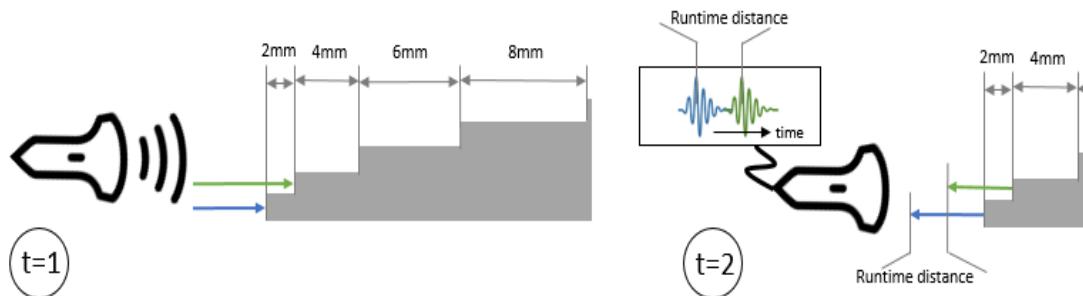
In order to test the application of image averaging, 25 images are acquired with the parameter specifications described in the lateral resolution test (Chapter 2.4.7), except for pulse parameter  $t_5$  which was reduced to 250.000 ns. The PLA phantom 1 (b) was placed at a 15cm distance from the transducer and scanned 25 times with the fixed set up without rearranging. It has three steps of 10 mm x 10 mm [height x depth] (see Figure 2.5, Chapter 2.1.5). The image data was averaged and analysed in a Python script and plotted with the **matplotlib** package.

#### *Determination of Axial Image Resolution*

In order to test the system's axial resolution, the 3D-printed phantom 2 (c) is used with steps of different depths. The depth differences of the steps are 2mm, 4mm, 6mm and 8mm (see Figure 2.16). Since the transducer is  $d = 30$  mm in diameter, it is assumed that the ultrasound beam is cylindrical and its width catches all five steps at once (phantom height 20 mm). This allows for recording all echoes in one single line. Acquisition parameters according to voltage test in Chapter 2.4.2 are utilised at 72V and 16MHz sampling.

The echo signal reflected at the surface of a phantom's step is a series of 5-7 oscillations at the pulse frequency of 1 MHz. Due to the fact that every echo in the signal consists of multiple

oscillations, the different echo signals from different steps can overlap. If the runtime difference is too small and the resulting echo trains are overlapping too far (i.e. the step depths are too small to make a difference in runtime for the echo), it effectively makes it impossible to distinguish between the two surfaces of the steps (see introductory Chapter 1.4.2, Rayleigh Criterion).

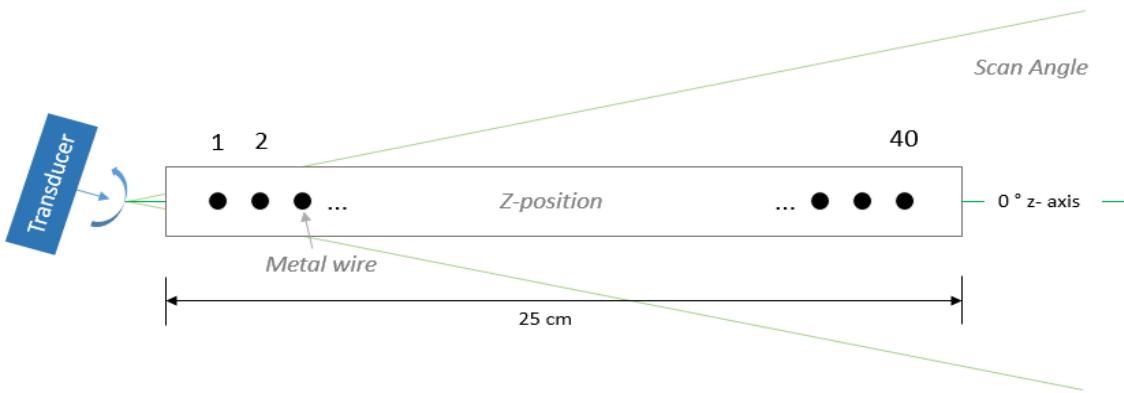


**Figure 2.16:** Step Phantom 2 (Figure 2.5 (b)) and its Runtime Distance

#### Determination of Lateral Image Resolution

To determine the impulse function of a transducer to high accuracy, the IEC recommends measurements using a hydrophone [37]. Since the measuring the ultrasound field of the transducer with a hydrophone is beyond the scope of this project this experimental test is simplified.

Using the experimental set up as seen in Figure 2.7, a 3 mm diameter metal wire was successively placed along the 0-degree z axis of the scan sector. The overall set up allowed to scan a distance up to around 25 cm. A small metal wire of 3mm diameter was perpendicularly placed to the direction of sound propagation. The Point Spread Function is determined by scanning the point source's impulse response, measuring it with the cylindrical ( $d = 30\text{mm}$ ) single-element transducer. For measuring the impulse response of the field, 40 measurements of different z-positions are taken respectively (see Figure 2.17). Starting with a position close to the transducer (approximately 45 mm), it is then moved by  $dz = 5\text{ mm}$  steps, in each position ultrasound image with an opening angle of 30 degrees and 44 lines in total (forward and backward rotation) and pulsing at 1 MHz with 72V at a sampling frequency of 16 MHz is acquired. Pulse train timings are equal to the default values described in Chapter 2.1.2.



**Figure 2.17:** Experimental Set Up for Determination of Point Spread Function

After the signal acquisition, images are reconstructed in MeVisLab with the following parameters stored into **parameters.json**: No TGC applied, FFT-band-pass-filter:  $f_0 = 0.8 \text{ MHz} / f_1 = 1.2 \text{ MHz}$ , down-sampling-factor: 16, Savgol-Window: 0.02, GaussKernelX: 1.5, GaussKernelY: 1. The images are saved into the experiment folder and loaded into a Python Script for further analysis. At this point the images are normalised to the range [0,1]. To determine the PSF for each z-position, the maximum in the image is determined and the respective line is extracted. With the **numpyamax()** function determining the z-position of the peak, the image row at position  $z$  represents the Point Spread Function of the scanned object in x direction (lateral).

To express the Point Spread Function not as experimental data but as a distribution, a Gaussian curve was fitted to all 40 PSFs. From a first glimpse at the data, a Gaussian distribution is suggested. The model fit was done by applying the **Imfit.Model** package which provides a **GaussianModel()** method for non-linear least squares curve fitting. From the Gaussian fits, the parameter  $\sigma$  is used for comparison of the PSFs along the z-axis.

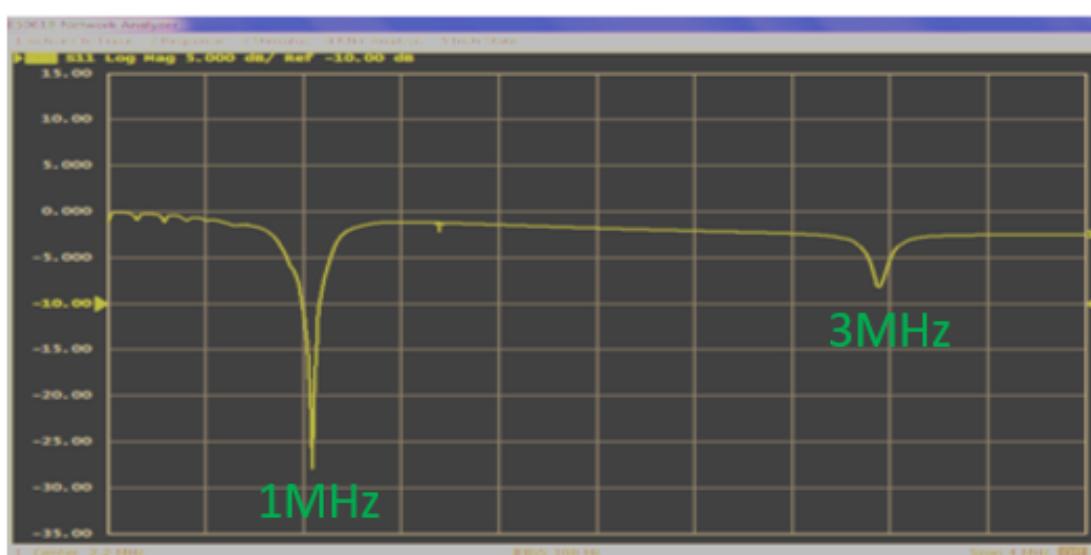
To improve the lateral resolution, the acquired images are subsequently deconvolved with their respective PSF in z-direction. The deconvolution of images is conducted in the free open-source software package “Octave” Version 5.1.0.0 [42]. From the “Octave Forge” collection of tools for several applications, the **signal** and **image** packages were used, providing a method **deconvwnr(image,filter)** for Wiener-filtering and **plot()** and **imshow()** methods for depicting the images. While the Wiener filter assumes a noise component, this is neglected here since the PSF is deconvolved with itself.

### 3. Results

The above described hardware components are successfully connected to form the experimental set up. With the illustrated Python routines which control the pulser voltage, the step motor rotation, the recorded signal transfer and procession and the image reconstruction, the LCUS can scan the maximum angle of 60 degrees in approximately 60 seconds. The length of acquisition time depends on the acquisition parameters which are entered into the graphical user interface, the speed of data transfer and the down-sampling factor. Tests described in Chapters 3.3 to 3.7 are conducted with the LCUS as implemented. Experimental testing and characterising the Low-Cost Ultrasound System renders the following results.

#### 3.1 Determination of the Transducer's Resonance Frequency

In Figure 3.1, the result of the network analyser test is shown. The first peak of the curve indicates a global maximum at approx. 1MHz with an amplitude of -28dB. This is the resonance frequency of the single-element transducer. If the transducer is operated at this pulse frequency, the performance of the piezo-electric crystal is optimal, i.e. the pulse emitted from the transducer in form of a pressure wave is the strongest at 1MHz. Sending a strong pulse results in being able to receive strong reflection signals. Hence, for the following experiments, a pulse length of  $t_5 = 500$  ns is chosen (pulse length = 1 / pulse frequency). The test furthermore identifies a second peak at approximately  $f = 3$  MHz with a significantly lower amplitude height of -5 dB. It suggests that operating the transducer at a pulse frequency of 3 MHz should be possible. However, it will yield weaker signals. Since 3 MHz is a harmonic frequency of 1 MHz, the reflection ultrasound signals from a pulse generated with 1 MHz will carry components oscillating at 3 MHz. This might allow higher harmonic imaging. This phenomenon is further explored in the Fast Fourier Transform spectrum analysis (Chapter 3.4).



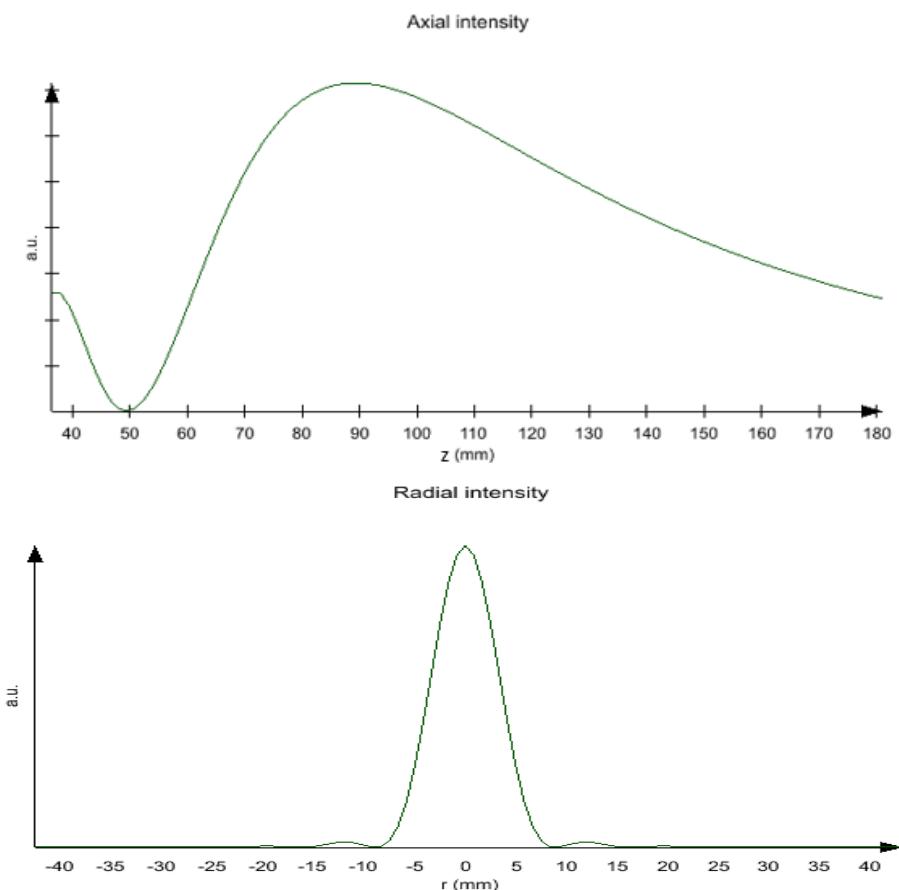
**Figure 3.1:** Network Analyser Test S1-1 Parameters of the Single-Element Transducer

### 3.2 Simulation of the Transducer's Ultrasound Field

Figure 3.2 shows the results of the ultrasound field simulation in axial (z) and radial (r) direction. The resulting absolute magnitude of intensity amplitude is not interpretable (depicted in arbitrary units). The relative behaviour, however, gives some indications.

Along its propagation axis (z-direction), near the transducer, the field intensity is weak, decreasing to 0 in intensity until  $z = 50$  mm. It then increases to a peak intensity at around 90 mm from the transducer. This represents the natural focus point of the beam which separated near- from far-field. Theoretically, the resolution of an ultrasound system is highest at this image depth z in the reconstructed image.

Looking at the lateral intensity distribution of the beam, it can be seen that the main lobe (central peak) amplitude is significantly higher than its first order side lobe peaks ( $x = \pm 11$  mm). The side lobes of the ultrasound beam can be neglected since they are not contributing to the reflected and recorded signal. The slice at 77% of the peak intensity (Rayleigh Criterion) value is approximately 10 mm wide. This represents the radial resolution achievable with the system without applying any further image processing.



**Figure 3.2:** Axial and Radial Intensity Plots of 30mm Diameter Piston Transducer rendered by MeVisLab Simulation Tool (Parameters:  $d = 30$  mm,  $f = 1$  MHz, in Water)

Since conventional ultrasound systems can achieve significantly higher resolution (see Chapter 1.1, compare with Figure 1.1) of 0.3 mm to 3 mm, a deconvolution filter is constructed in Chapter 3.7.

### **3.3 Analysis of Un0rick Board's Sampling Frequency and Voltage**

As described in the materials section, the Un0rick board provides the opportunity to digitally sample the received analogue signal at 16 MHz, 24 MHz, 48 MHz or 64 MHz. An increase in sampling frequency results in an incline in generation of data points. This extremely expands the time it takes to reconstruct an image. A jump from 16 MHz to 64 MHz translates into a factor 4 in both x and z - direction and thus a factor 16 in pixels requiring processing in the aftermath. Hence, quite the opposite operation of down-sampling the recorded signals is applied in the implemented signal processing routine (see Figure 2.11) speeding up the imaging process. Furthermore, as the transducer used for the experimental tests operates at the low frequency of 1 MHz and higher frequencies are eliminated by the band-pass filter anyway (see next Chapter 3.4), the sampling frequency can be kept low at 16 MHz without any considerable loss of information.

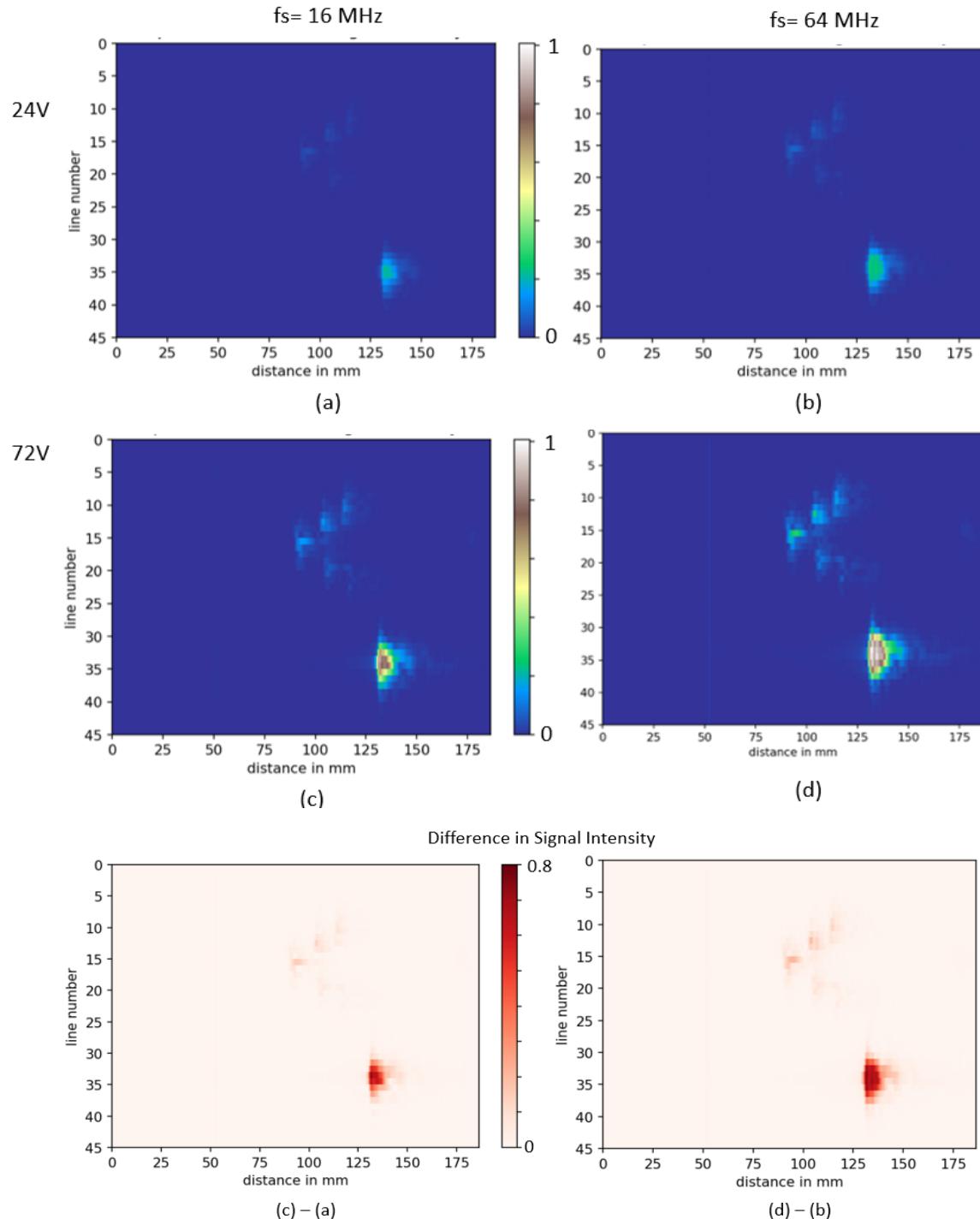
Figure 3.3 depicts the resulting “Flat 2D images” of the voltage test. Images (a) to (d) are normalised to the maximum value of all four images and plotted with the same colour-map to make them comparable. The acquired 45 single lines are written into a row each. These are not the true spatial reconstructions of the experimental set up.

The difference in signal intensity between the images taken at 16 MHz and 64 MHz are calculated by subtraction of (c)-(a) and (d)-(b). Furthermore, the maximum values of each image are calculated which are each found at the centre of the reflected echo signal of the metal object. The maximal value of image (a) is 0.23, image (b) 0.26, image (c) 0.89 and image (d) 1 in arbitrary units.

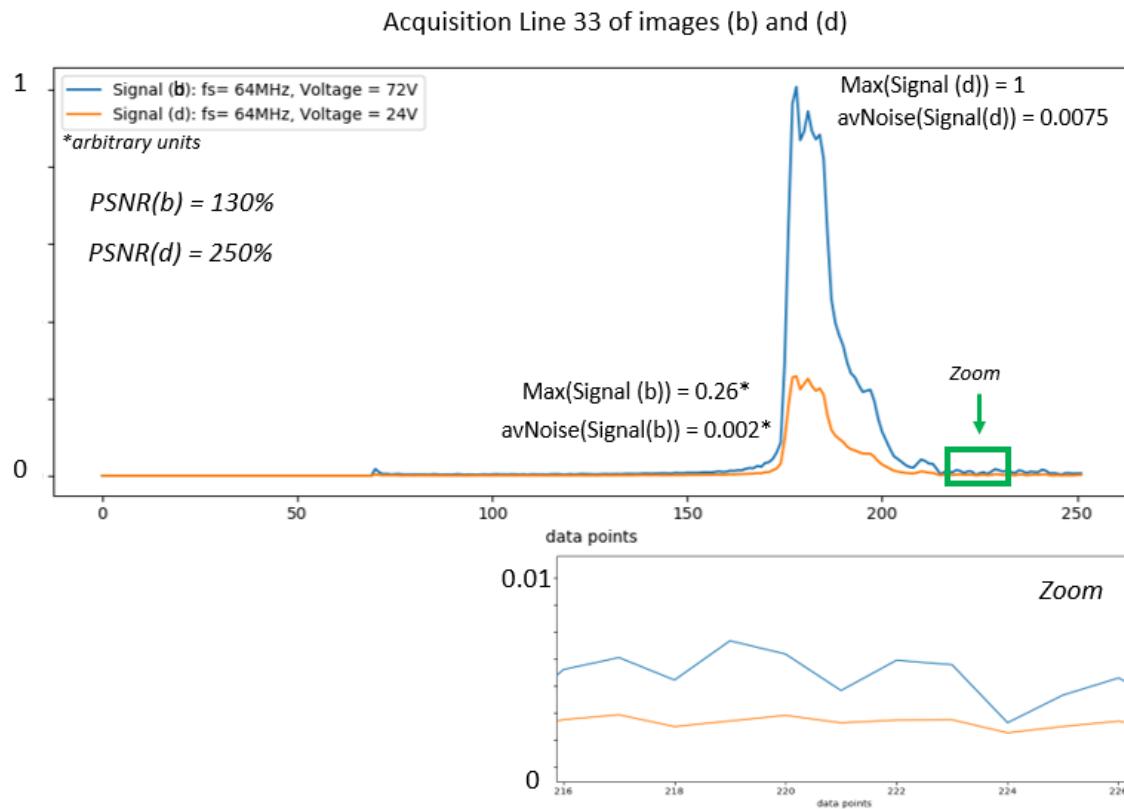
The difference in received echo signal is therefore a *factor 4* for the metal object when *tripling the pulse voltage*. For the reflection of the PLA surfaces, this still amounts to an increase by factor 1.5 to 2. With this experiment, one can clearly perceive the difference in acoustic impedance: Metals reflect much stronger echoes than plastics.

Taking a closer look at the acquisition line 33 of images (b) and (d), in which the peaks are found, the difference in image quality with reference to the PSNR is apparent. To investigate this, the PSNR was approximated by dividing the peak value of the echo signal by its average noise value. Even though the occurring noise increases with the voltage, the SNR improves. Comparing the images acquired with a sampling frequency of 64 MHz, one can state that the PSNR is improved by factor 2 when the voltage applied to the transducer is increased threefold. A similar behaviour is found for the comparison of recorded images (a) and (c).

Concerning the imaging properties of the Un0rick board, it is justified to acquire signals at the lowest sampling frequency of 16MHz. To reduce the PSNR and therefore improve the image quality, pulsing at the highest frequency of 72V is optimal.



**Figure 3.3:** Results of Voltage Test: Images at 24V (a) & (b), 72V (c) & (d) Difference Images



**Figure 3.4:** Voltage Intensity Plot of Acquisition Line 33 from Images (b) and (d) (Peak Values of both Images are found in this Line) and a Zoom in on the respective Noise

### 3.4 Signal Fast Fourier Transformation Frequency Spectrum Analysis

Figure 3.5 depicts results of the single line's frequency spectrum analysis. The unfiltered signal line (a) recorded echo signals which are sitting at approximately 80 ns and 200 ns. Due to its surrounding noisy behaviour (50 to 250 ns) and the occurring systemic oscillations (0 to 70 ns), it is difficult to distinguish from the undesired components of the signal. With its amplitude peak, the highest point of the systemic oscillation at 1 a.u., the PSNR only amounts to a weak 30%. Therefore, in order to improve the peak signal to noise behaviour of the Low-Cost Ultrasound System, Fast Fourier Transform filtering is applied to the recorded signal.

Pulsing at frequency of 1 MHz, the desired echo signals ought to carry a frequency of 1 MHz. The high-pass filter's cut off frequency is therefore set to just under 1 MHz ( $f_0 = 0.8$  MHz). The application of the filter immensely improves the signal quality as seen in Figure 3.5 (b). With this step, the systemic oscillations at the beginning of the signal are eliminated, rendering a much higher peak signal to noise ratio of around 90%. The filtered signal is drawn in light blue and plotted over the original signal, colour-coded in purple, for comparison.

Applying a band-pass filter ( $f_0 = 0.8$  MHz and  $f_1 = 1.3$  MHz) to the recorded signal shows even better results. It suppresses unwanted high frequency noisy behaviour in the area of 120 ns to 200 ns (see Figure 3.5 (c), zoom).

Furthermore, the possibility of *harmonic imaging* with the system was explored. Within the analysed frequency range, harmonic oscillations of first and second order (2 MHz and 3 MHz) might occur. As the previously described network analyser test of the transducer spectrum (see Chapter 3.1) suggests, especially at 3 MHz echo signals should be taken into account. The FFT-band-pass filter was hence set to  $f_0 = 2$  MHz and  $f_1 = 4$  MHz. Analysing the FFT spectrum of the signal shows peaks at 1.1 MHz, 2.2 MHz and 3.3 MHz. The resulting band-pass filtered signal (light blue, Figure 3.5 (d)) registered all echo signals except the last one (at 240 ns). The signals are of lower amplitude (that means intensity) by factor 4 compared to Figure 3.5 (c). It also detects the second characteristic echo signal (at 90 ns) with significantly lower amplitude compared to the first. These have the same height in amplitude when applying the band-pass filter in Figure 3.5 (c).

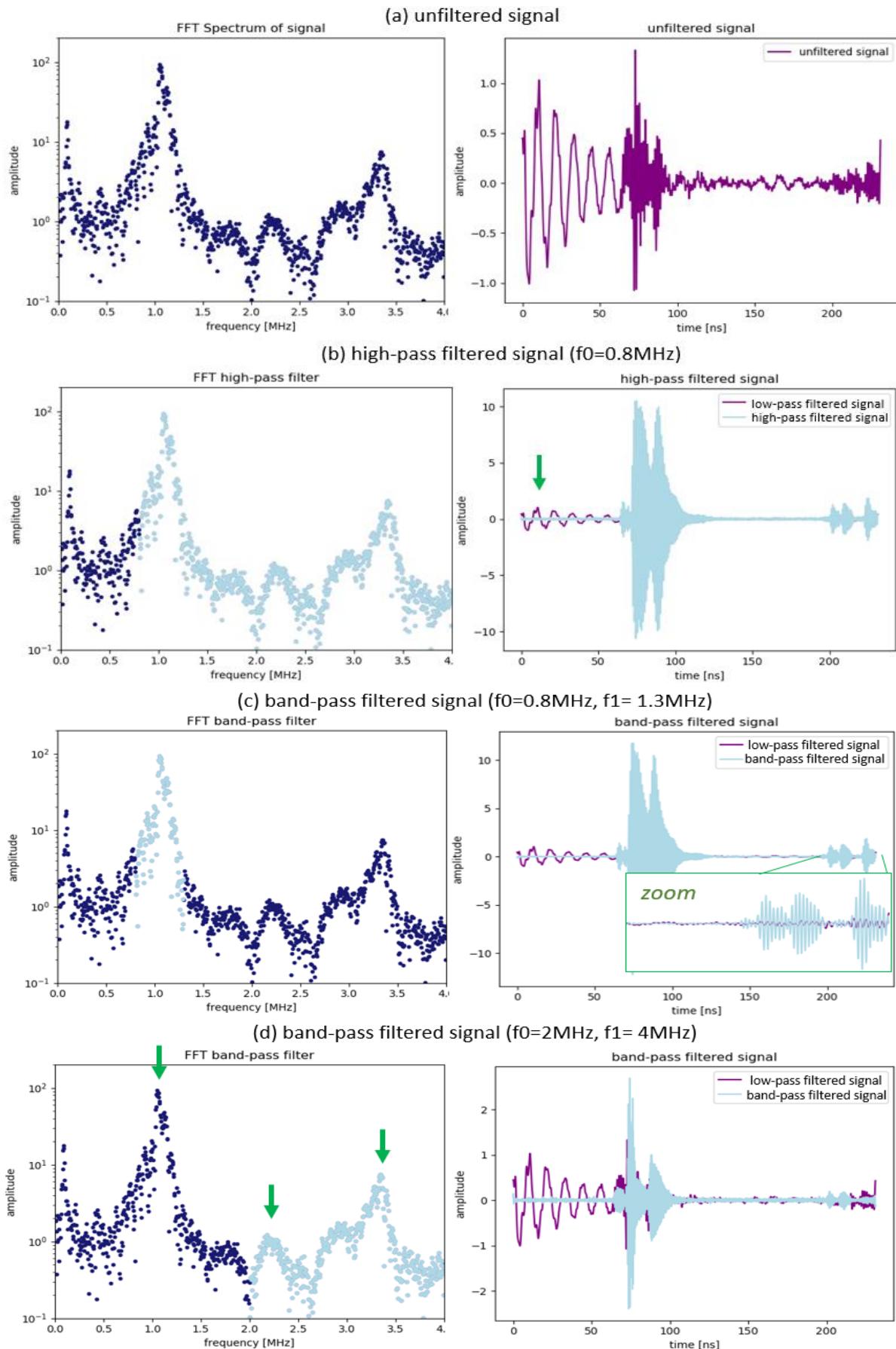
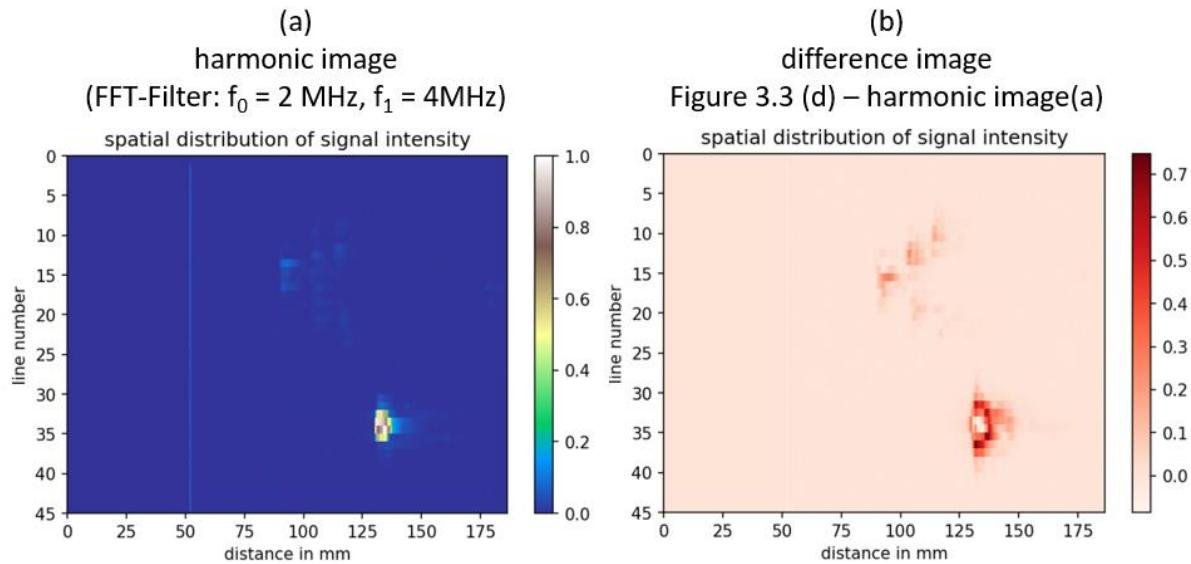


Figure 3.5: Fast Fourier Spectrum Analysis of one Recorded Signal Line

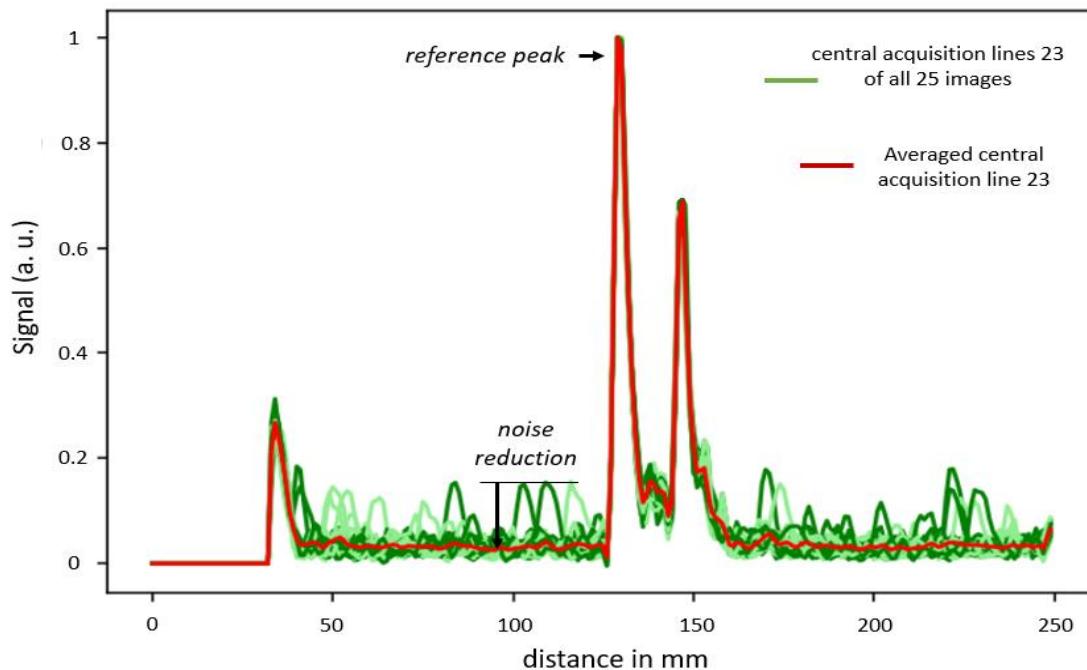


**Figure 3.6:** (a) Harmonic Image of Figure 3.3 Image (d) with FFT-filter Range [2 MHz, 4 MHz] and (b) Difference Image of Figure 3.3 Image (d) and Harmonic Image (a) from Figure 3.3

In order to further explore the behaviour of the LCUS concerning harmonic imaging, an image from the previous voltage test (Figure 3.3 image (d), 72 V) was filtered in a range of 2 MHz to 4 MHz, allowing 3 MHz harmonic components of the signal (suggested by S11 Parameter Test, Chapter 3.1) to appear in the image. In Figure 3.6 (a), one can clearly perceive both structures, the PLA step phantom 1 and the metal plate. The signal intensity however is generally weaker than the original image, steps of the phantom less defined. This is confirmed by the difference Figure 3.6 (b). The intensity of the remaining signal is lowered by 70 % around the metal plate phantom and 30 % around the PLA phantom by using only the harmonic frequency components of it. The white spot at the centre of the metal plate signal show no loss in signal intensity in the difference image. These are the two image lines where the metal plate's surface is orthogonal to the transducer beam. In the ultrasound beam simulation, it was shown that this central line of the beam has the highest intensity (Figure 3.2, Chapter 3.2). This result suggests that if an object's surface is positioned orthogonal to the transducer surface, the intensity generated by the transducer is great enough to reflect harmonic components. If this condition is not given, pulsing at 72V and 1 MHz does not generate enough signal intensity to conduct harmonic imaging.

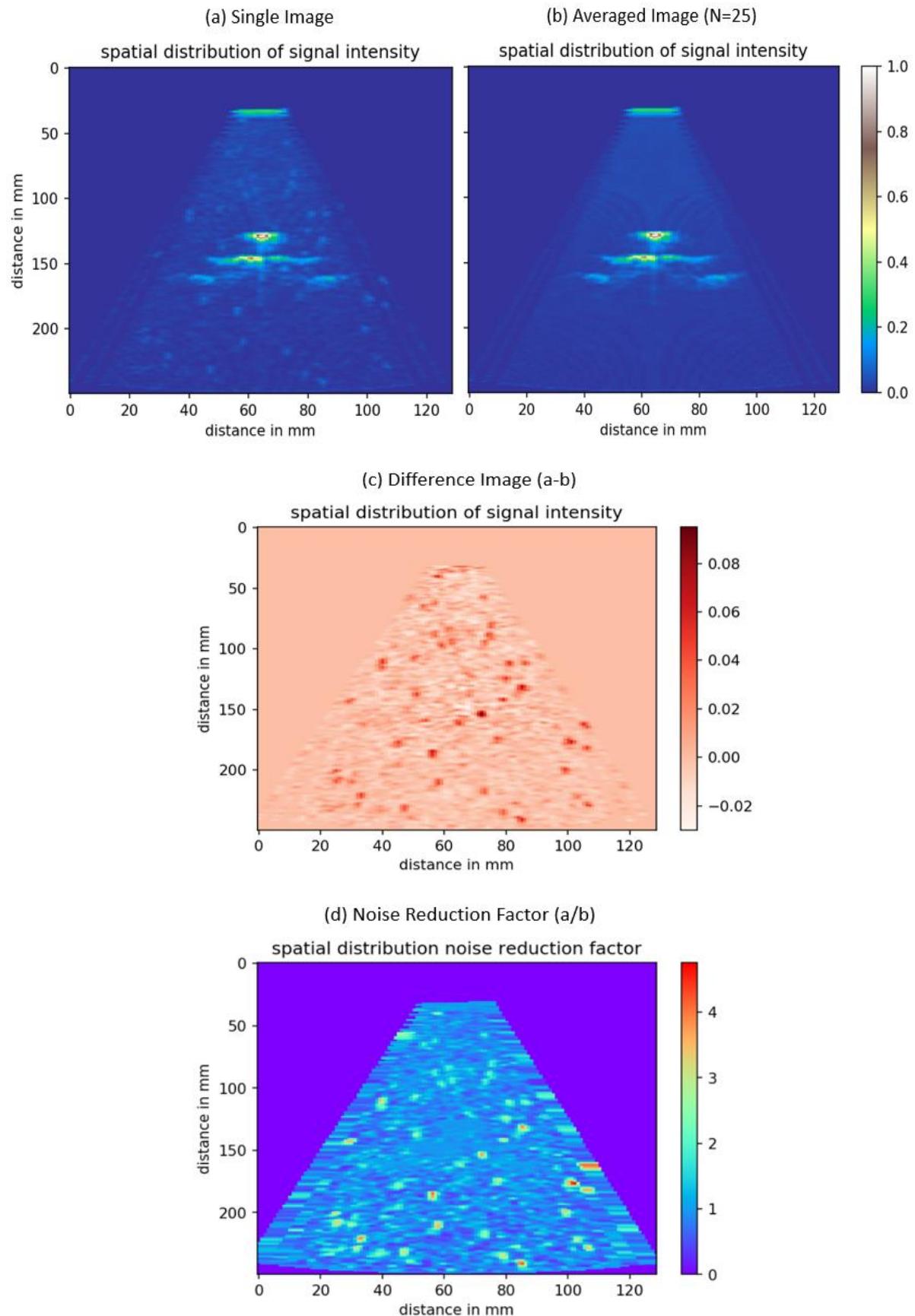
### 3.5 Application of Image Averaging

Figure 3.7 depicts the central lines (line 23) of each 25 acquired images in light and dark green. The first peak in the signal (at 40 mm) represents the systemic oscillation which could not be eliminated by the filtering or the cropping. The averaged signal of all 25 images is displayed in red. Representing the two echo signals reflected by two steps, the second and third peaks have a difference in depth of 10 mm. The signals are normalised to their respective highest peaks. Peak values of the noisy behaviour from the individual image lines to (40 to 125 mm and 160 to 250 mm) range up to 0.15. Its overall pattern is random. The averaged signal has a mean value of around 0.04 in this region. The averaged signal shows clearly that it approximates the wanted echo signal peaks and the undesired systemic oscillation quite well, whereas the random noisy behaviour is cancelled out.



**Figure 3.7:** Envelopes of Central Signal Lines of 25 Acquired Images (green) and the Respective Averaged Signal (red)

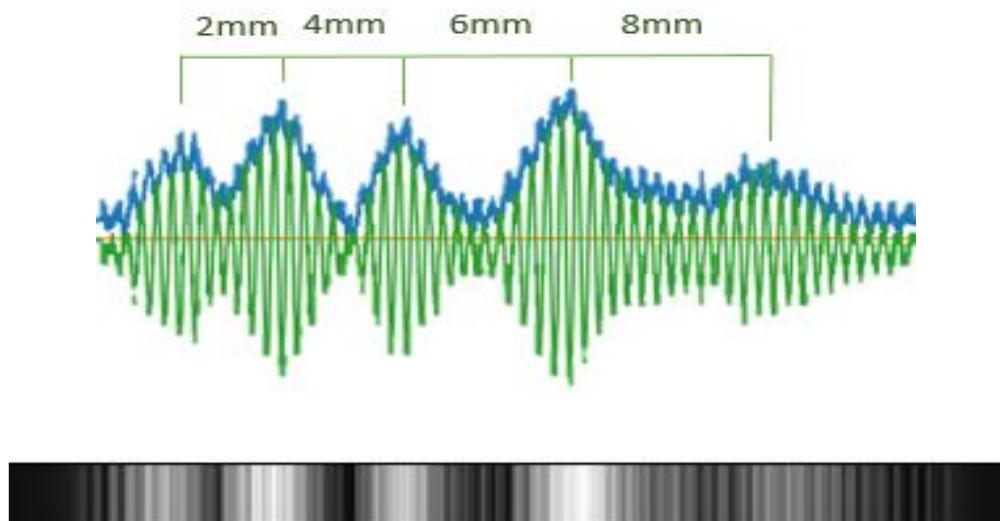
The visual effect of averaging an ultrasound image can be observed in Figure 3.8. in which a single acquisition image (a) carries a lot of speckle behaviour around the scanned object. This is eliminated in the averaged image (b), only the echo reflection of the desired object remains. The generated difference image (c) shows the noise pattern removed from the single image by applying averaging. In order to quantify the improvement in image quality (Peak signal to noise (PSNR)), a noise reduction factor map was calculated. It shows the absolute value of the single image divided by the averaged image. In some spots, the PSNR is improved by factor 4. For a large part of the image, the noise is reduced by factor 1.5 to 2.



**Figure 3.8:** (a) Single Image, (b) Averaged Image  $N = 25$ , (c) Difference Image  $(a - b)$ ,  
(d) Noise Reduction Factor for  $N = 25$   $(a/b)$

### **3.6 Determination of Axial Image Resolution**

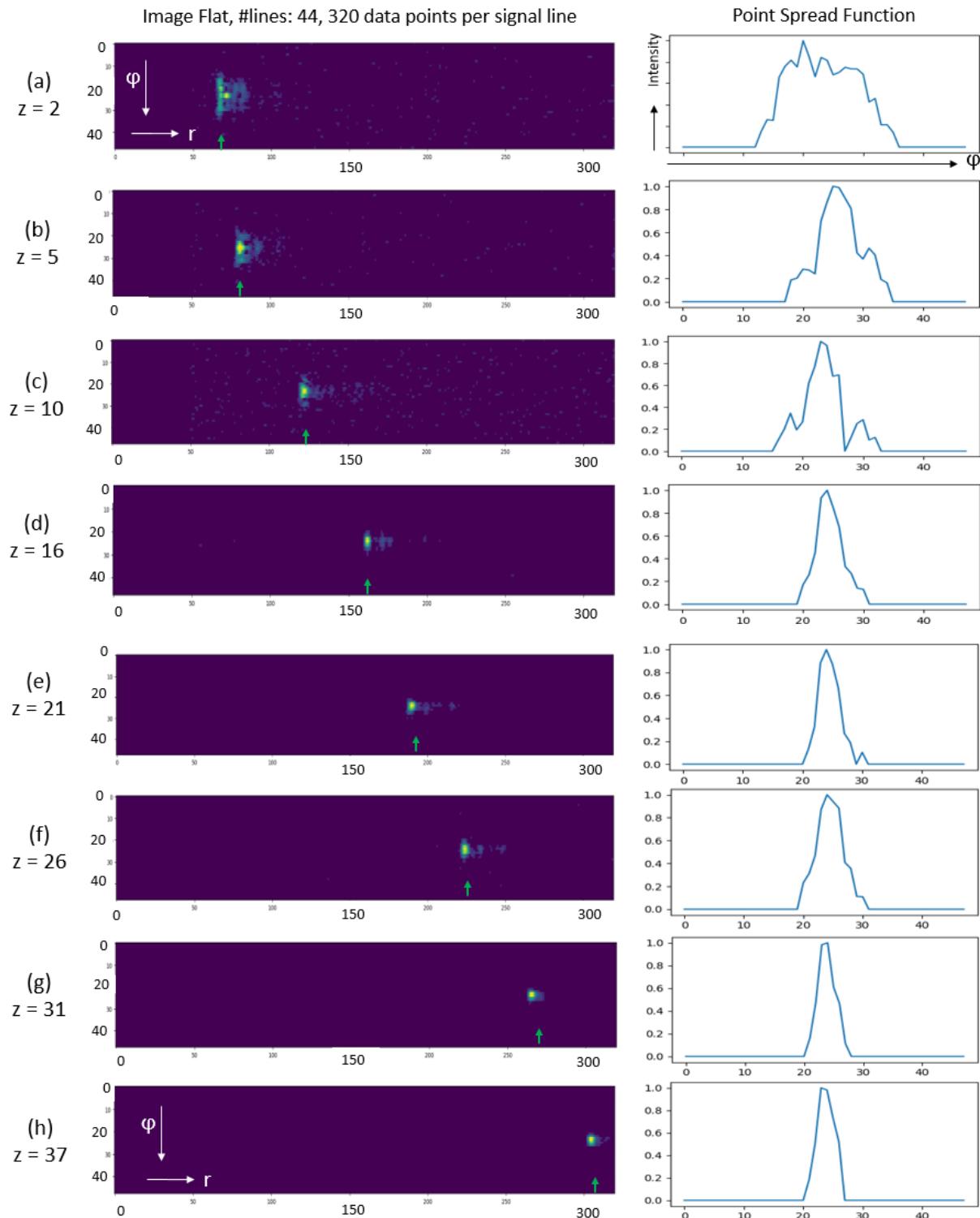
In Figure 3.9, one can see the result of the axial resolution test. In the signal, the first two peaks can clearly be identified as two individual echoes. These are echoes from the first two steps of the PLA-step phantom 2, Figure 2.5 (c), with a distance of 2mm. Plotting the signal into a single line array of grey values, the four individual white image points are distinguishable. It can therefore be deduced that in a reconstructed grey-value ultrasound image, the two individual surfaces with 2 mm distance are clearly identifiable. Therefore, this Low-Cost Ultrasound System is capable of achieving an axial resolution of 2mm pulsing at 1 MHz at 72V along the beam.



**Figure 3.9: Recorded Echo Signal of Axial Resolution Test**

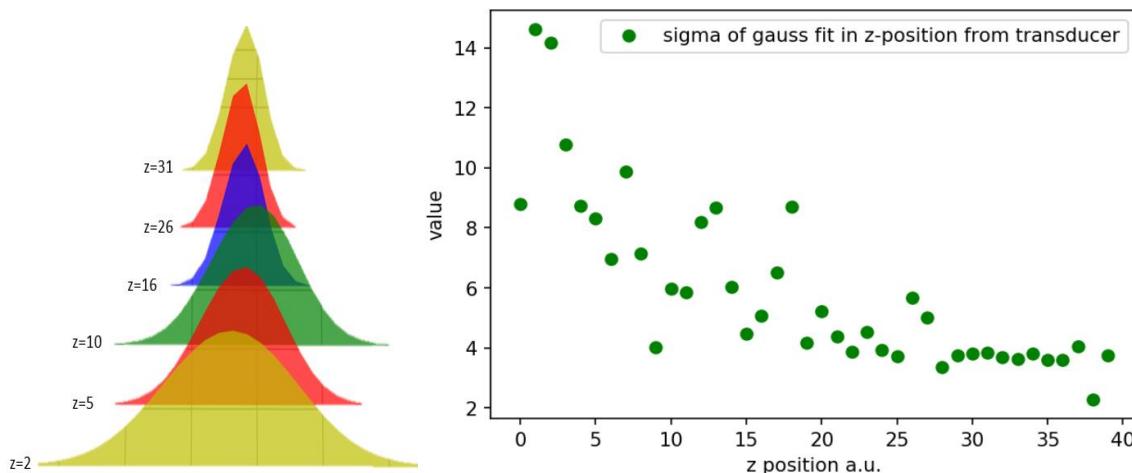
### 3.7 Determination of Lateral Image Resolution

Figure 3.10 shows the reconstructed “Flat 2D image” of 8 z-positions 2,5,10,16 21, 26, 31 and 37 and their respective Point Spread Functions. These are depicting the image slice intensities (normalised) at the position of the green arrows.



**Figure 3.10:** Flat Acquired Images of a 3 mm Point Source in (a) Z-Position 2, (b) 5, (c) 10, (d) 16, (e) 21, (f) 26, (g) 31 and (h) 37 and the respective Point Spread Functions in the  $r$  Slice (along  $\varphi$  axis)

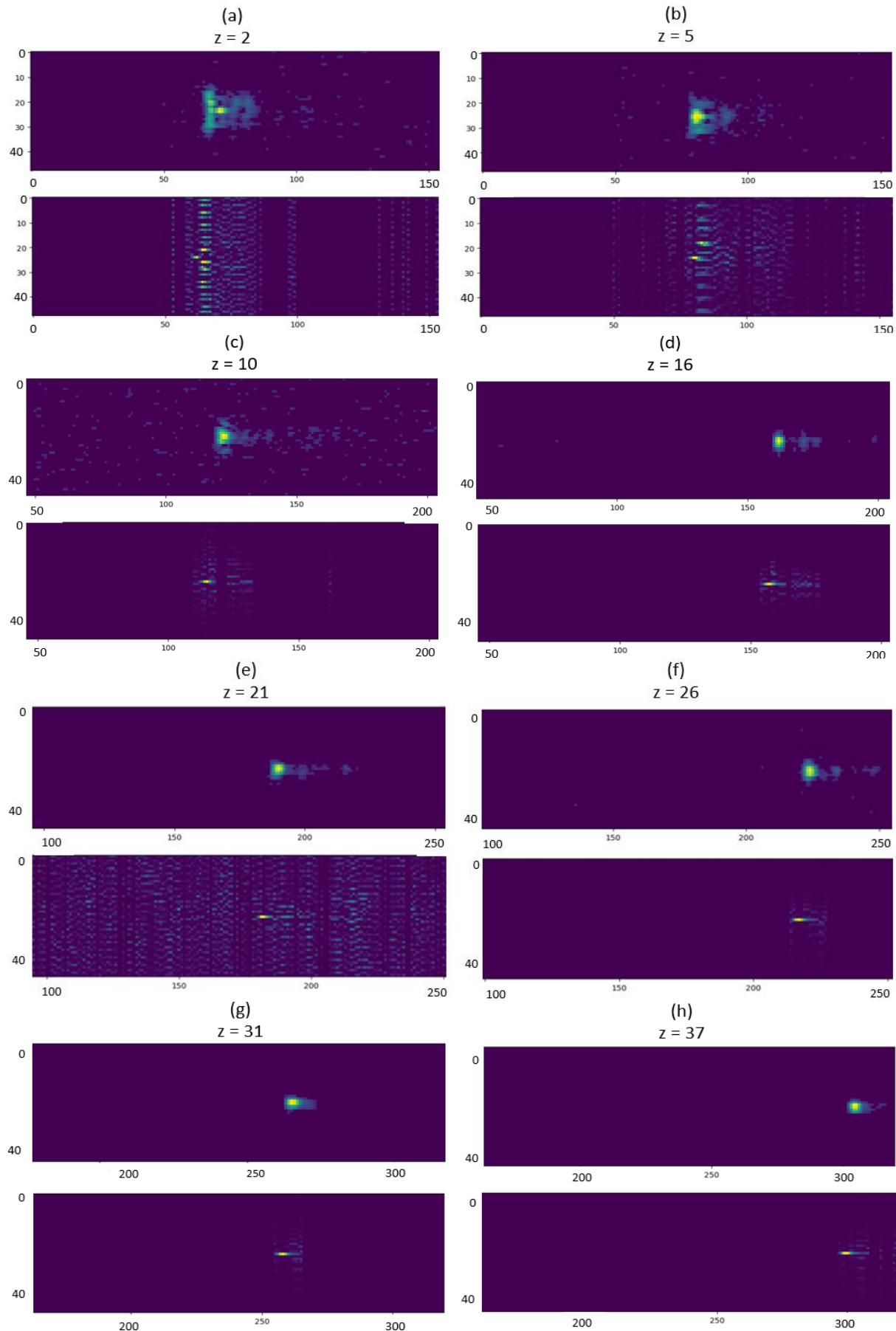
In “Flat 2D Image” (a), the point object is still positioned in the near-field, very close to the transducer (approx. 5 cm from the transducer’s surface). Here the signal reflected from the metal wire is spread over 25 acquisition lines. This can be seen in the SPF  $\varphi$ -axis extracted and shown on the right to each “Flat 2D Image” in Figure 3.10. At  $z = 5$ , the SPF is still quite spread out (18 image lines). At  $z$ -position 10, the representation of the point object seems to become more distinct, where 7 image lines (20–27) form a distinct and steep peak. According to the simulation of the ultrasound field (Chapter 3.2), this is the region of the natural focus point (approx. 9.5 cm from the transducer). Imaging the point object further from the transducer, it is scanned by less acquisition lines. In the images  $z = 31$  and  $z = 37$ , it can be presumed to converged to 5 image acquisition lines.



**Figure 3.11: Gaussian fits of several z-positions and  $\sigma$ -values for 40 z-positions**

Fitting a Gaussian curve over each of the PSFs produces an instrument to compare the width of the spread - as the  $\sigma$  – value of the Gaussian model always incorporates the same ratio (68 %) of probability mass for every curve. From Figure 3.11, it is apparent that the above described decline in number of acquisition lines recording a signal from the point-like source is of exponential decaying nature. From  $z$ -position 10 onwards, the decline can be described by a linear descent. According to the theory on piston transducers (see Figure 1.5, Chapter 1.4), the spread width of the ultrasound beam should be equal to its diameter, then decline to half its diameter into a focal point and thereafter incline again. However, from this experimental, no focal point can be identified. The decay is monotonous.

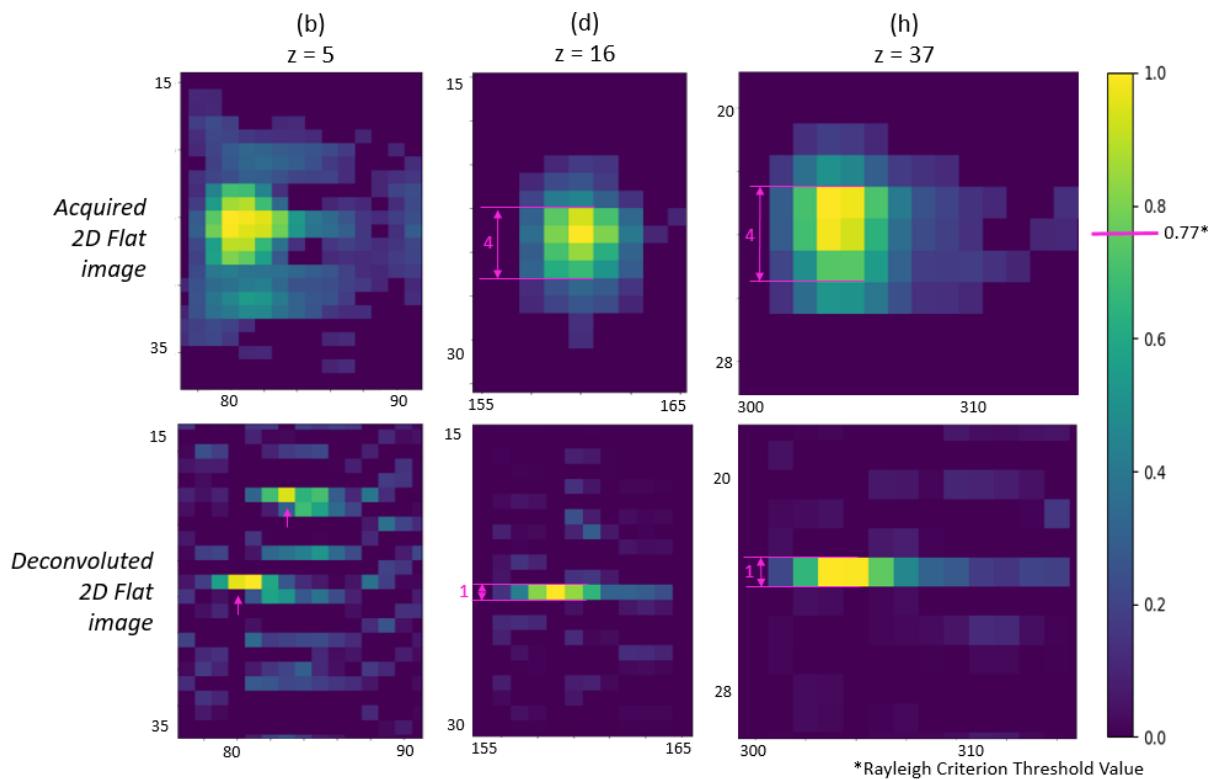
With the reconstruction algorithm used for the true spatial position image (True 2D Flat), a point object placed 25 cm from the transducer would be theoretically fanned out to 12 mm ( $\sin(1.4/180\pi) * 250\text{mm}$ ) in lateral spread if scanned by 5 acquisition lines (as in  $z = 31$  and  $z = 37$ ). Reasoning from the width of the transducer ( $d = 30\text{mm}$ ), the lateral resolution ought to be between 15 mm and 30 mm. From the True 2D reconstructed images, this could be experimentally confirmed (Figure 3.14, top left). The selected five z-positions displayed in Figure 3.14 are of a lateral width of 20 to 35 mm.



**Figure 3.12:** Deconvolved “Flat 2D Images” of (a) Z-Position 2, (b) 5, (c) 10, (d) 16, (e) 21, (f) 26, (g) 31 & (h) 37

In order to improve the lateral image resolution, the images were deconvolved with their respective Point Spread Functions. Figure 3.12 shows the resulting deconvolved “Flat 2D images” for z-positions (a)-(h). Only the relevant section of 150 data points around the point object is depicted in each image.

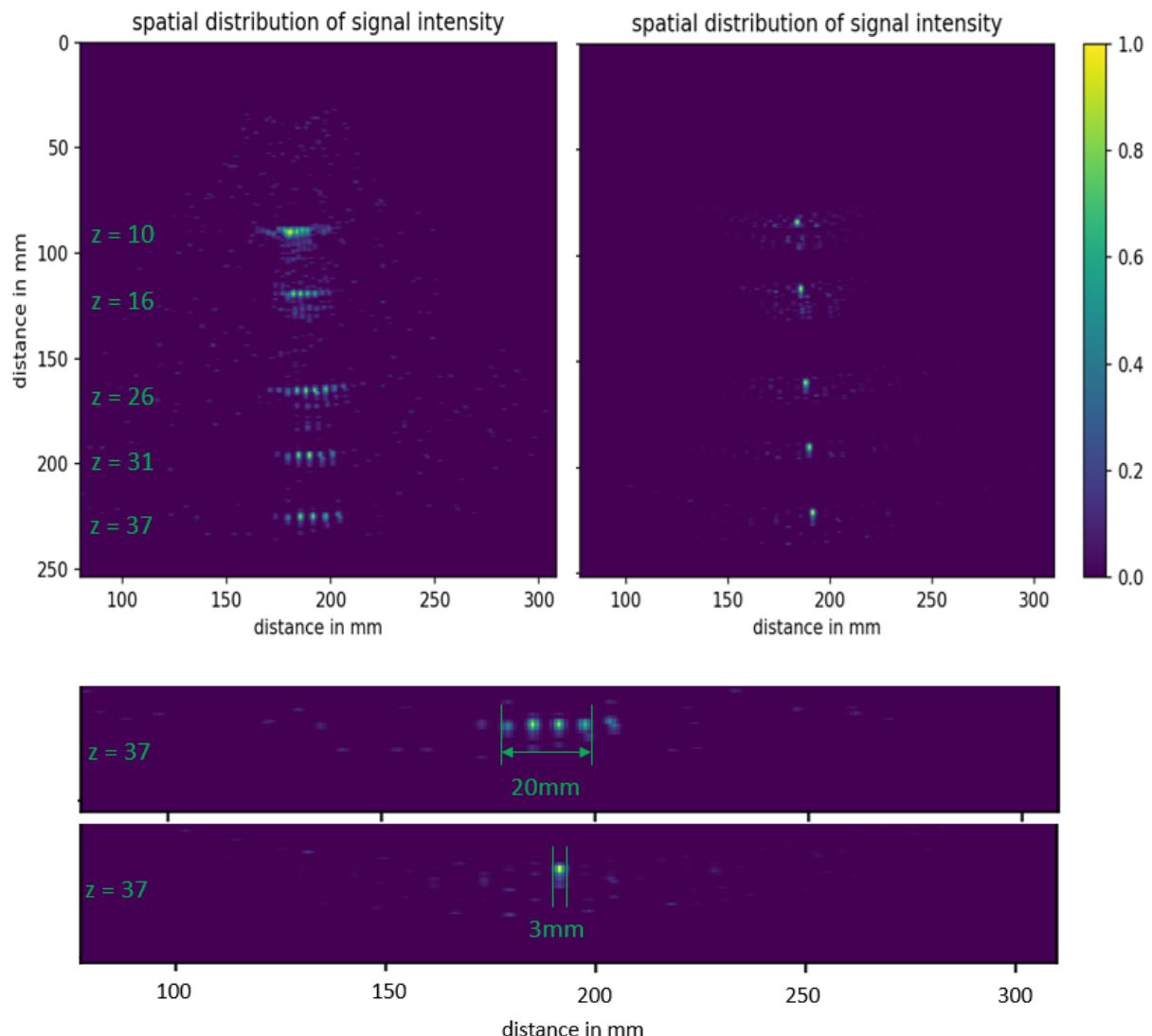
For the first two “Flat2D images”, the deconvolution did not work satisfactory since it returned an image with artefacts and no clearly identifiable point object. In Figure 3.13, image (b), the results could be falsely interpreted as two small point objects as the pink arrows in zoom on image (b) point out. For the overall interpretation of images (c) to (h), it can be stated that deconvolving the images with their respective PSF reduced the pixels depicting signal for the scanned points objects significantly. If applying the Rayleigh Criterion (compare equation (3), Chapter 1.4) Value of 0.77 as threshold to identify the object, four image lines are reduced to one by the deconvolution as indicated in Figure 3.13. For images where the representation of the metal wire is more distinct, the result of the deconvolution also shows less artefacts. Comparing images (c), (e) and (f) highlight the fact that noisy behaviour is also reflected in the deconvolved images with artefacts in proximity to the object. Image (e) is inadmissibly noisy.



**Figure 3.13:** Zoom on Z-position (b), (d) and (h)'s point source object before and after the Deconvolution

In order to finally see the spatial effects of the deconvolution, the “True 2D Images” are reconstructed from the deconvolved “Flat 2D Image” arrays (algorithm can be found in Annexe). The results are shown in Figure 3.14. For compact depiction of the results, the five image arrays (c)  $z = 10$ , (d)  $16$ , (e)  $26$ , (g)  $31$  & (h)  $37$  are added and plotted into one image. The 3 noisy images (a), (b) and (f) are not reconstructed.

From the top right deconvolved image, one can tell that the 3mm wire object is displayed as 5 equal sized point-like white spots. Indeed, they are positioned along one axis. The noise seen in the image on the top left is reduced and only occurs near the point objects. A zoom on  $z = 37$  shows the lateral spread reduction from approximately 20 mm to 3 mm with the deconvolution. By applying the computational method of deconvolution to the post-processing of acquired ultrasound signals with the LCUS, a resolution improvement of approximately 700 % can be achieved.



**Figure 3.14:** Sum of 5 Deconvolved Reconstructed “True 2D Images”, Z-positions (c)  $z = 10$ , (d)  $16$ , (e)  $26$ , (g)  $31$  & (h)  $37$  above and Zoom on  $Z = 37$  below

## 4. Discussion

Subject to research in this project is the implementation and characterisation of the LCUS. In order to derive answers to the posed research questions, its three elements of development so far described - *the hardware, the software architecture and the experimental tests* – require discussion.

### 4.1 What are the technical capabilities of the Low-Cost Ultrasound system?

The online support structure provided by Luc Jonveaux, initiator of the Un0rick project, and the freely available open-source software of the Un0rick Board - in combination with a Raspberry Pi - allow to build a Low-Cost Ultrasound System with limited resources concerning time, budget and expertise. Using Python 3 for code implementation ensures compatibility between all platforms (Windows, Raspberry Pi and Un0rick Board) and simplicity for the developer as it is widely used in scientific programming. Since all open-sourced components and software used in this project are published under a general public license (GPL), the provided scripts could simply be adjusted to the requirements of the project. Generally, it can be stated that the system is modular which allows for easy further development.

#### *Experimental Set Up*

With this rotational set up, a scan angle of up to just under 120 degrees and an acquisition line length of around 25cm could be covered by one scan. This area and dimension is sufficient for application to the human body. Due to the radial set up, image information located distant from the transducer is sparse which is partly be encountered by applying a Gaussian -filter of small (x,z)-kernel size and rotating twice (forward and backward) shifted by one step.

The step motor's performance used in the set up is limited since its build form only allows rotational steps of 1.4 degrees (2 step units), failing when programmed to execute single or half steps. Moreover, it is lacking a mechanism to measure its accuracy. It is not possible to judge if it actually executes the rotation as reliably as assumed.

Also, phantoms are placed in the polar coordinate system “by hand”. This means that spatial/distance information in the reconstructed image cannot not be assessed in sub-mm accuracy. For all conducted tests, the experimental set up was stable and test is reproducible when not rearranging its components.

### *Developed Software and Graphical User Interface*

With the implemented software architecture, ultrasound images can be acquired. Signal and image procession achieve 2D images. The developed GUI leads the user through the acquisition process.

The raw signal storage and data transfer after the acquisition is implemented by generating `.json` files. The upside of using this string format is that is human readable and therefore debuggable. The transfer, however, is slower than other binary file formats which require less bytes to store the information.

Creating a graphical user interface for using the Low-Cost Ultrasound System's software makes it easier to understand and less prone to erroneous usage. Furthermore, by introducing a web socket connection module in MeVisLab, the signal processing and image reconstruction could be executed on a Windows 10 PC, resulting in faster imaging than if using the RP for this purpose.

In the software code lies potential for speeding up the image acquisition time for one image which at this stage takes 90 seconds. As all code is written in Python, the scripts are easily adaptable by anyone wishing to continue developing the system.

### *Experimental Tests*

The single-element transducer's pulse frequency of 1 MHz generates distinct echo signals and is used throughout experiments. With the LCUS it is possible to scan simply-shaped objects such as PLA - phantoms and metal plates and depict these in 2D ultrasound images.

Testing the transducer's frequency range showed a resonance frequency at 1 MHz. The fact that the FFT spectrum analysis shows a peak at 3.3 MHz as a harmonic oscillation of the pulse, suggests that the optimal operating frequency is closer to 1.1 MHz.

The simulation of transducer's ultrasound field with properties 3 cm-diameter and 1 MHz pulse frequency, indicates that the side lobe intensity of the beam is so low that it can be neglected. It is therefore justified to approximate the Point Spread Function with a single Gaussian curve representing the main beam. Furthermore, the simulation reveals the spread width of the ultrasound beam to be approximately 11 mm. This is an unacceptable dimension for resolution on ultrasound images (compare Figure 1.1). Hence post-processing is required to improve image quality. This is done by applying deconvolution with its PSF-filter.

Using the UnOrick board at its highest possible voltage of 72 V, as expected, resulted in better PSNR behaviour than operating it at lower voltages (24 V). It is justified to down-sample recorded signals without loss of information in the reconstructed ultrasound image.

The analysis of the frequency spectrum illustrates the need to apply a FFT-Filter. A high pass-filter cutting off just under the pulsing frequency carries the largest improvement in PSNR and is

necessary. A band-pass filter improves the SNR even further but simultaneously excludes harmonic frequency components of the signal. Harmonic imaging is possible but results in weaker echo signals and therefore higher SNR compared to the filters which include the frequency of 1 MHz.

The averaging test proves that the PSNR of the ultrasound image is significantly improved in some regions by factor 4, generally by factor 2. It does not, however, achieve the theoretical noise reduction factor of 5 (applied  $\sqrt{K}$ , Chapter 2.4.8) when averaging 25 images.

The theoretical focal point of the ultrasound beam could not be identified in the data. This might be due to the rotational experimental set up, not rendering enough signal information in sparsely represented regions of the image. Also, due to time constraints the ultrasound field could not be measured with higher accuracy with an appropriate measuring device (hydrophone).

By determining the Points Spread Function of the system, the computational method of deconvolution can be applied to the acquired images. Hereby, a *sevenfold* improvement of lateral resolution in images could be achieved. The  $d = 3$  mm wire (point object), which is smeared to 20 mm in lateral direction by the system, could be reconstructed to a depiction of a 3 mm point in the ultrasound image after the deconvolution. During the project, the convolution was performed on a laptop PC and took around 120 seconds per image. If executed on a more performant computer or with parallelising algorithms, it could be done faster.

Comparing the finally achieved lateral resolution of 3 mm to conventional ultrasound systems (compare Figure 1.1), it can be stated that the LCUS is well within the acceptable range.

#### *Further Limitations and Methodological Critique*

The transducer was only operated at the lower range of medical ultrasound frequencies (at 1 MHz). This pulse frequency is not applicable to most medical applications. These require higher frequencies of up to 15 MHz.

According to the IEC, experiments should be conducted in degassed and distilled water at a temperature of 25°C and a level of reproducibility should be incorporated [37]. These conditions could not be replicated. The temperature was not monitored and the water not degassed and distilled.

Furthermore, the TGC behaviour is advised to be investigated [38] in order to characterise an ultrasound system which was not done in this project.

Taking the PSNR as a measure is a simple approach to assess the image quality. However, there exist better suited image quality metrics [39] to judge from a human visual perspective.

Using Octave to deconvolve the acquired image interrupts the chain of automatic signal procession since the data has to be loaded into the Octave environment manually.

## 4.2 Which Implications for the Health Care Market can be derived?

The dominantly used hand-held device “VScan” General Electric [GE Healthcare GmbH, DE, Solingen] is available at a price of just under EUR 3.900 without a transducer element (or a 76 EUR monthly fee). Another comparable ultrasound device is the “Butterfly iQ” [Butterfly Network, US, CT, Guilford] which has yearly fee for usage of 1.800 EUR. Even conservatively estimating the cost of the LCUS amounts to much smaller total sum of approx. 537,50 EUR. It can therefore be stated that the system used in this project comes at a significantly lower price than existing systems on the market. At the moment, the Un0rick ultrasound board is produced on a small batch size of under 10 pieces. If produced on a larger scale, the price ought to be significantly lower due to scaling effects. Assuming the price for the LCUS could be reduced by 20 % is pushing the price well under the 450 EUR mark, resulting in a price of 15 % of the cheapest available HHD.

In Figure 4.1, a comparison of conventional HHDs with the LCUS developed in this project considers further characteristics. The LCUS can generate a pulse voltage of 72V which is satisfactorily close to the 90V typically used for ultrasound pulses. In order to cater to the full range of medical ultrasound pulse frequency conventionally used, the single-element transducer can easily be replaced by another with a broader band-width transducer. The 3 mm resolution achievable with the LCUS is on the upper range of conventional ultrasound systems. This might be sufficient for certain medical applications and not acceptable for others. Whilst conventional HHD are IP-protected and the service for adaption of the systems - if allowed - is not free, the LCUS does not succumb to any of these restrictions. Due to the open-source GPL and the free support structure, it is quite on the contrary, designed for adaptation.

	<i>Conventional HHD</i>	<i>Low-Cost US</i>
<i>Pulser Voltage</i>	Ultrasound devices typically use a pulse generation voltage of up to 90V [42],	24V, 48V and 72V
<i>Pulse frequency</i>	1 MHz to 15 MHz	1 MHz, 3MHz (harmonic)
<i>Time to Image</i>	Real - time	60 seconds
<i>Resolution</i>	0.3 mm to 3mm [Figure 1.1]	3 mm (after deconvolution)
<i>Price</i>	4.000 -10.000 EUR	537.5 EUR
<i>Adaptability</i>	Patent (IP-Protected) Maintenance Service (costs)	GPL Free Support Online Structure

**Figure 4.1:** Comparison of Properties commonly used HHD and LCUS

The image quality (axial and lateral resolution, PSNR behaviour) is - at this stage of development – not quite up to standard with already existing systems. With the help of implementing computational imaging methods such as averaging and FFT- and deconvolution-filters, the results show that a considerable improvement in imaging quality can be achieved. Further techniques which can be undertaken to enhance the (image) quality of the system are described in the concluding outlook on future research.

Within the technical performance limitations of the low-cost system, several applications in medical imaging can be identified considering the literature review on Hand Held Devices (HHD) in the introduction, Chapter 1.1. Evidently, if adjusted to very specific examinations such as echocardiography and the determination of the aortic diameter, a simple device is “good enough” for the purpose at hand. Furthermore, the use of such frugal ultrasound device in middle and low income countries especially in natural disaster situations, where no high-end ultrasound systems are available, is promising. As laid out in the introductory Chapter, ultrasound as an imaging modality in medicine has its advantages as investments in devices and training are lower compared to CT and MRI. Especially for applications remote from hospital infrastructure, it is extremely suitable.

Recent advantages in computing power lead the shift from physical performance of ultrasound devices to computationally heavy signal and image post-processing in ultrasound imaging. Using artificial intelligence (AI) and deep learning methods in the processing procedure to enhance image quality, reduces the technical requirements of the ultrasound device needed for achieving state-of-the-art ultrasound images. In combination with AI and cloud computing, simple devices with limited technical capabilities such as the *Low-Cost Ultrasound System* are expected to be capable of rendering images of comparable quality than more expensive ultrasound systems. Computational power is generally less costly than the development and maintenance of physical hardware.

The synergy of the modular software architecture as well as the hardware architecture with its open-sourced environment offers the extension of the system by such methods. Free of charge Python packages are already available for the implementation of e.g. neural networks for machine learning. Using a conventional 3D-printer for rapid prototyping of the transducer holder and phantoms, shortens the development time conventionally needed to set up a prototype. Due to the inexpensiveness and the availability of all components, the ultrasound system therefore ought to be used in a research context.

With this low investment in physical resources required, it offers the chance to use it for *student and research projects and other educational purposes* as anticipated by the developers of both Un0rick ultrasound board and the RP mini-computer. It is suggested to be used in the training of medical engineers and software developers which need to learn how ultrasound systems work. In general, the inexpensiveness of the system lowers the hurdle for users to experiment with and develop applications in medical ultrasound. The open-source documentation and support infrastructure assures the accessibility to understanding the ultrasound system.

### *Embedding the Results into Existing Fields of Research and Mapping the Technology*

Studying the surrounding literature on simple ultrasound devices developed with inexpensive components and open-source concepts, several recent and emerging fields of research in Technology Innovation Management are worth mentioning with respect to the LCUS.

*Open Innovation.* Often cited, the innovation funnel by Chesborough [44] indicates, that - in order to successfully innovate - modern companies should not just generate ideas from within, but also from sources outside of the organization. Open-sourced projects take this idea even further by creating a core product and allowing the public to use or even further develop it, to draw on the “wisdom of the crowds”. Research [44] affirms that particularly in technologies associated with *science, medicine and education*, organisations should fund the development of open-source hardware development if interested in maximizing the return on public investments. From an open innovation perspective, it is indeed recommendable for the UnOrick team to continue the development of its hardware ultrasound boards as an open-source and collaborative project.

The term *Frugal Innovation* was first coined by Tiwari et al. [46] and describes the phenomenon of increasing development of products with a focus on inexpensiveness, robustness and reduced technological complexity, “good enough” for specific applications. Characteristics of frugal products are functionality, easiness-to-use, low price and scalability. These technologies are typically found in Low and Middle Income Countries, stemming from the need for “good enough” simple technologies which can be used without the infrastructure required for high-end solutions. In a “qualitative analysis of the potential socio-economic impact”, Tiwari et. Al [46] monitored and identified a growing importance of frugal devices in the German industry (since 2010). In light of this field of research, the over-engineering of modern high-tech solutions especially for medical devices is questioned, since it leads to higher costs with respect to initial investment and training required. As described in the Introduction Chapter 1.1, many stakeholders such as the World Health Organisation (WHO) and the Federation of Societies for Ultrasound in Medicine and Biology (EFSUMB) [32] & [34], recognise ultrasound imaging as a technology offering solutions especially - but not exclusively - in LMIC. Portable ultrasound devices therefore fall into this category of frugal innovation products, with a growing impact on and attractiveness to “high-tech” markets such as the German health care market.

A growing field of interest led by Carlis Y. Baldwin (Harvard Business School, [47]) with respect to product design, is the concept of *Modularity*. The study of *Modularity* of a system investigates to which extent it is decomposable and reconfigurable [47]. A Modular system is flexible, its single units “modules” are exchangeable with a unit of the same functionality without having to replace the neighbouring ones in the process. This concept is commonly used in software development. It can extend to hardware, physical systems as well. Applied to the LCUS, it means that single units such as the micro-controller can be exchanged for another one (Arduino by Raspberry Pi) both still utilising the same interfaces (Python scripts, USB) without too much effort. A high degree of modularity is beneficial during the research and development phase of a system lowering the hurdles to constantly reconfigure it [47].

#### **4.3 What are future research applications of the system?**

Some approaches to enhancing the Low-Cost Ultrasound System are sketched for future research.

##### *Improvement of Current System*

As briefly discussed above, the experimental set up requires improvement. The used step motor should be replaced by a more reliable one. Implementing appliances with which the accurate rotational behaviour and placement of phantoms can be traced, as well as ensuring a more precise determination of spatial information in the experimental set up, is recommendable. This would add reproducibility and reliability to experimental testing.

Some properties of the ultrasound board such as the use of time gain compensation (TGC) and the possibility to use two separate piezo-elements connected to the TX and RX channel of the UnOrick were not used. Operating, the device at the lowest sampling frequency of 16 MHz was only theoretically argued. The effect of these parameters should be experimentally investigated in order to further characterise the system.

Since for most medical applications a higher frequency of 2 to 16 MHz is required to receive enough signal, other transducers with higher resonance frequencies should be tested. Increasing the frequency additionally improves the image resolution and could lead to better imaging quality when using harmonic imaging.

The developed software should be revisited with respect to acquisition time optimisation by replacing data formats Python packages with more suitable ones. Furthermore, deconvolution ought to be implemented with a Python routine instead of octave, so that the process can be automated.

### *Extension of Current System*

Averaging the acquired images has been proven to reduce noise in the ultrasound image and therefore improving image quality. The development of an acquisition sequence which includes a window of  $K$  images to be averaged should be explored. In this analysis,  $K= 25$  was tested. Due to the inverse quadratic behaviour, a lower number of images should be implemented, which will still result in a significant improvement of SNR (factor 3 if  $K= 9$ ). To not slow down the acquisition time of an ultrasound image, a display method needs to be derived. This could be implemented in the form of a sonar monitoring display which shows old data until new scans are available.

To accelerate signal and image processing, the advantages of the MeVisLab software should be integrated. The software suite offers a plethora of signal processing methods coded in C++. Capitalising on the possibility to use modules which execute calculations in C++ instead of Python, will speed up the imaging process.

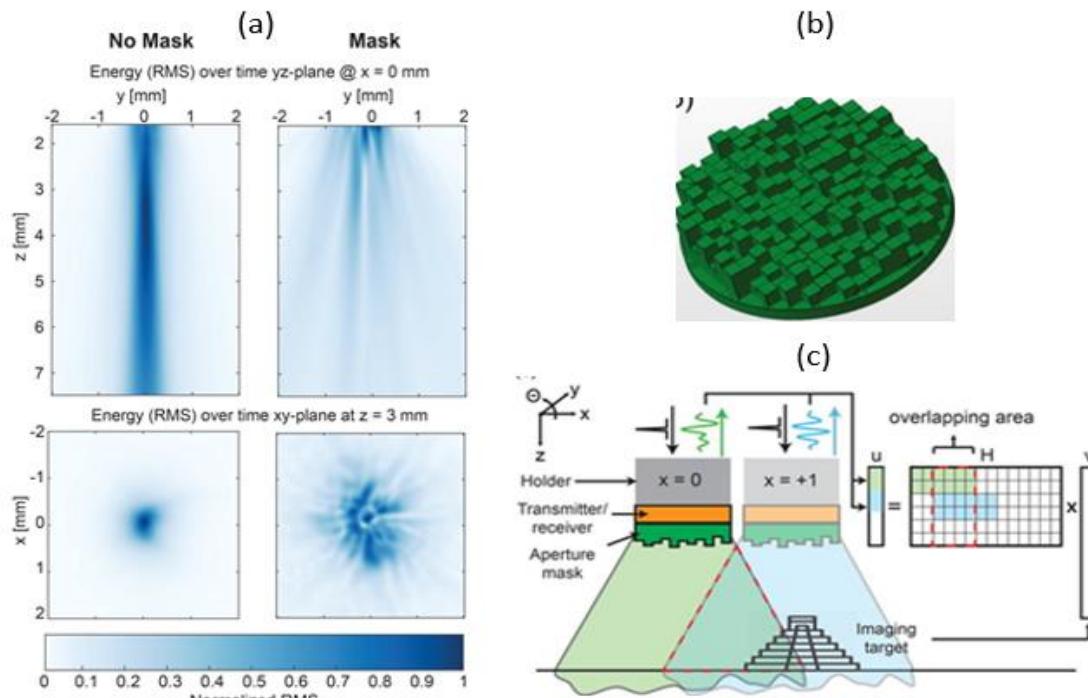
The most recent version of *RP4 B* incorporates a Wi-Fi unit allowing to connect to the mini-computer. For the LCUS, it was not used, since it was launched nearing the end of the project (end of June 2019). For future projects, the RP 4 B's Wi-Fi interface should be used to transfer the parameters and the raw data replacing the web-socket connection. This would make the system truly portable. The ultrasound device could then be used from remote, where the user only requires an internet connection of some sort. Implementing this feature follows the train of thought of using the device outside of clinics and other infrastructural rich environments (e.g. a research lab).

### Phase Encoding Masks

Another technique to achieve high resolution images using a single-element ultrasound transducer system was proposed by Janjic et. al [48]. The Dutch team used 3D-printed (inexpensive) phase-encoding masks (depicted in Figure 4.2 (b)) which are placed in front of the emission surface of the piezo-element and combined this with high performance computational image reconstruction. With its pattern (in propagation direction), the aberration mask imposes its phase structure on the beam, breaking the spatial coherence between voxels at equal acoustic propagation delay (as seen in Figure 4.2 (a)). A priori, having measured the single-elements transducer's impulse response (ideally with a hydrophone) for each voxel, allows to allocate the recorded echoes to a scatter map. The decorrelation of the ultrasound field achieves isotropic resolution at all depths of the beam not just at the focal point. An image is reconstructed by solving the inverse mathematical problem described by model [48]

$$u = H v + n \quad (6)$$

Where  $u$  is the recorded image,  $H$  the measured impulse response matrix of the ultrasound field and  $n$  is the additive noise component, solved for  $v$  via a LSQR algorithm. When scanning an object as seen in (c), much more spatial information can be extracted resulting in higher resolution images.



**Figure 4.2:** (a) Ultrasound field transmitted from the sensor without & with the aperture mask  
 (b) phase encoding mask (c) schematic of the imaging procedure [48]

Extending the LCUS by the use of phase-encoding masks and implementing the described computational image reconstruction process should be explored in future research on this project. The inexpensiveness and simplicity of the method aligns with the project goal.

## 5. Conclusion

Many stake-holders, such as the WHO and the German Federal Statistic Bureau, have identified the need for inexpensive ultrasound imaging. Influencing factors such as *rising costs* for diagnostic imaging examinations in the German health system, the existing *call for simple and low-cost ultrasound devices in remote areas* (e.g. in Low- and Middle Income Countries) as well as emerging innovation research in the fields of *modularity, open-source* and *frugality* affirm and underline the motivation for researching the LCUS.

This work represents a *proof of concept* for this inexpensive open-source ultrasound system. The implementation of the LCUS entails three elements of development. Firstly, hardware is set up and connected to form a space for experimental testing. Secondly, the existing open-source software is reviewed, further ultrasound signal processing and image reconstruction methods are developed and a GUI is built. Thirdly, experimental tests to explore the technical capabilities of the system are conducted. This includes the investigation of FFT-filtering, studying the effects of averaging and assessing the image quality by determining the image resolution.

### *What are the system's technical capabilities?*

Within the limited resources available to set up the LCUS, the experimental allowed to test its properties. With an ultrasound pulse of 1 MHz and a pulser voltage of 72 V, 3D-printed step phantoms and a metal plate and wire could be distinctly depicted in the reconstructed image. The implemented FFT-filter significantly improves the recorded signal quality. Image averaging additionally ameliorates image resolution. The method of deconvolution with a measured PSF-filter shows promising results for enhancing the ultrasound image resolution with post-processing methods. The inclusion of MeVisLab software allows to build a graphical user interface. The modular architecture of the software, the use of Python as a coding language and its open-sourceness, ensure the easiness to adapt the system for the developer.

### *Which implications for the health care market can be derived?*

Comparing its properties to conventional ultrasound device's, it is not at level with existing standards concerning *image quality*. However, its advantages are apparent if considering its *adaptability* and *costs*. In combination with rapid-prototyping technologies such as 3D-printing and the online support provided, the open-source concept of the LCUS proves the easiness to use and develop it further. The system exhibits that with inexpensive components it has a multitude of future applications in medical ultrasound while being less costly than existing hand-held devices on the market. It could play a role in ultrasound research projects and provide a solution for the call for frugal "good enough" ultrasound devices.

### *What are future research applications of the system?*

To leverage its potential, future research should focus on further development of the software methods and the optimisation of its hardware components. Averaging and deconvolution filtering should be implemented in the signal processing routine and the use of Wi-Fi to transfer data ought to be realised. Furthermore, the symbiosis of 3D-printed phase-encoding masks and post-processing power of the Low-Cost Ultrasound System should be explored.

## 6. Bibliography

- [1] GENESIS Platform, Statistisches Bundesamt/ Federal Statistical Office Germany [online]. Available: <https://www-genesis.destatis.de/genesis/online/>. [Accessed: 30. November 2019, 14:22]
- [2] Destatis. "Fallpauschalenbezogene Krankenhausstatistik (DRG-Statistik) 2016. Diagnosen, Prozeduren, Fallpauschalen und Case Mix dervollstationären Patientinnen und Patienten in Krankenhäusern," *Destatis*, 2016. [Online]. Available: [https://www.destatis.de/DE/Themen/Gesellschaft\\_Umwelt/Gesundheit/Krankenhaeuser/Publikationen/Downloads-Krankenhaeuser/fallpauschalen-krankenhaus-2120640167004.pdf?\\_\\_blob=publicationFile&v=4](https://www.destatis.de/DE/Themen/Gesellschaft_Umwelt/Gesundheit/Krankenhaeuser/Publikationen/Downloads-Krankenhaeuser/fallpauschalen-krankenhaus-2120640167004.pdf?__blob=publicationFile&v=4). [Accessed: 29 November 2019, 13:09]
- [3] T. Szabo, *Diagnostic Ultrasound Imaging: inside out*. Elsevier Academic Press, 2017.
- [4] "How Much Does An MRI Cost," *How Much Does An MRI Cost | GE Healthcare*. [Online]. Available: <http://www.gehealthcare.co.uk/feature-article/how-much-does-an-mri-cost>. [Accessed: 1 November 2019, 22:41]
- [5] A. Jung, C. Schmieder, M. Schmidt, A. Holle, „Was kostet die Sonografie – Ergebnisse einer Kosten- und Prozessanalyse an einem Versorgungskrankenhaus“. Nürnberg: *Ultraschall in Der Medizin*, 2008. Available:  
[https://www.researchgate.net/publication/245887795 Was kostet die Sonografie - Ergebnisse einer Kosten- und Prozessanalyse an einem Versorgungskrankenhaus](https://www.researchgate.net/publication/245887795_Was_kostet_die_Sonografie_-_Ergebnisse_einer_Kosten-_und_Prozessanalyse_an_einem_Versorgungskrankenhaus). [Accessed: 28 October 2019, 15:34]
- [6] A. Rykkje, J. F. Carlsen, and M. B. Nielsen, "Hand-Held Ultrasound Devices Compared with High-End Ultrasound Systems: A Systematic Review," *Diagnostics (Basel, Switzerland)*, 15-Jun-2019. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6628329/>. [Accessed: 27 October 2019, 17:53]
- [7] V. Galusko, O. Bodger, and A. Ionescu, "A systematic review of pocket-sized imaging devices: small and mighty?" *Echo Research and Practice*, pp. 113–138, 2018. [Accessed: 21 October 2019, 10:54]
- [8] N. Cardim, H. Dalen, J.-U. Voigt, A. Ionescu, S. Price, A. N. Neskovic, T. Edvardsen, M. Galderisi, R. Sicari, E. Donal, A. Stefanidis, V. Delgado, J. Zamorano, and B. A. Popescu, "The use of handheld ultrasound devices: a position statement of the European Association of Cardiovascular Imaging (2018 update)," *European heart journal cardiovascular Imaging*, 01-Mar-2019. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pubmed/30351358> . [Accessed: 23.10.2019, 16:48]
- [9] D. M. Becker, C. A. Tafoya, S. L. Becker, G. H. Kruger, M. J. Tafoya, and T. K. Becker, "The use of portable ultrasound devices in low- and middle-income countries: a systematic review of the literature," *Wiley Online Library*, 10-Jan-2016. [Online]. Available: <https://onlinelibrary.wiley.com/doi/full/10.1111/tmi.12657>. [Accessed: 27 October 2019, 18:02]
- [10] World Health Organisation Geneva. "Future Use of New Imaging Technologies in Developing Countries. Technical Report Series No. 723. Report of a WHO Scientific Group, pp. 67, 1985.

- [11] "Vscan mit Dualsonde: GE Healthcare: Ultraschallgerät im Taschenformat," *GE*. [Online]. Available: <https://www.ge-ultraschall.com/vscan-portfolio/vscan-mit-dualsonde-2-2/> . [Accessed: 23 October 2019]
- [12] L. Jonveaux, "Arduino-Like Development Kit for Single-Element Ultrasound Imaging," *Journal of Open Hardware*, vol. 1, no. 1, 2017. Available: <https://openhardware.metajnl.com/articles/10.5334/joh.2/> . [Accessed: 25 September 2019, 10:36]
- [13] L. Jonveaux, "Murgen: An Open-Source Ultrasound Imaging dev-kit side project," *GitHub*, 09-Aug-2017. [Online]. Available: <https://github.com/kelu124/murgen-dev-kit/> . [Accessed: 10 June 2019]
- [14] Z. Taylor, L. Jonveaux, C. Caskey, "Development of a Portable and Inexpensive Ultrasound Imaging Device for Use in the Developing World." *Young Scientist Journal*, Vanderbilt University, 2017. Available: <http://youngscientistjournal.org/youngscientistjournal/article/development-of-a-portable-and-inexpensive-ultrasound-imaging-device-for-use-in-the-developing-world> . [Accessed: 20 October 2019, 16:42]
- [15] "Raspberry Pi Foundation Strategy Paper 2018-20." [Online]. Available: <https://static.raspberrypi.org/files/about/RaspberryPiFoundationStrategy2018–2020.pdf>. [Accessed: 30 June 2019, 12:22]
- [16] "Wie funktioniert Ultraschall?," *Stiftung Gesundheitswissen*. [Online]. Available: <https://www.stiftung-gesundheitswissen.de/gesundes-leben/koerper-wissen/wie-funktioniert-ultraschall> . [Accessed at: 15 October 2019, 15:59]
- [17] "Fundamentals of Ultrasound," *Radiology Key*, 18-Aug-2016. [Online]. Available: <https://radiologykey.com/fundamentals-of-ultrasound/> . [Accessed: 23 October 2019, 10:23]
- [18] A. Ng and J. Swanevelder, "Resolution in ultrasound imaging," *Continuing Education in Anaesthesia Critical Care & Pain*, vol. 11, no. 5, pp. 186–192, 2011
- [19] "The principle of basic ultrasound". [Online], Available: [www.echoxedia.org/index.php?title=The\\_principle\\_of\\_ultrasound](http://www.echoxedia.org/index.php?title=The_principle_of_ultrasound). [Accessed: 2 November 2019, 16:34]
- [20] "Rayleigh Criterion: CAS CMS," *Rayleigh Criterion / CAS CMS*. [Online]. Available: [https://astronomy.swin.edu.au/cms/astro/cosmos/R/Rayleigh Criterion](https://astronomy.swin.edu.au/cms/astro/cosmos/R/Rayleigh%20Criterion). [Accessed at 13:50, 30 September 2019]
- [21] Y. Paul, D. V. M. Barthez, and Peter, "SIDE LOBES AND GRATING LOBES ARTIFACTS IN ULTRASOUND IMAGING," *Wiley Online Library*, 23 May 2005. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1111/j.1740-8261.1997.tb02104.x> . [Accessed: 11:47, 1 October 2019]
- [22] O. T. von Ramm and S. W. Smith, "Prospects And Limitations Of Diagnostic Ultrasound Proc. SPIE 0206, Recent and Future Developments in Medical Imaging II, 26 December 1979,[Online], Available: <https://doi.org/10.1117/12.958184>, [Accessed at 13:01, 30 September 2018]

- [23] C. Dalitz, R. Pohle-Frohlich, and T. Michalk, “Point spread functions and deconvolution of ultrasonic images,” *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 62, no. 3, pp. 531–544, 2015.
- [24] N. Rao, “Experimental Point Spread Function of FM Pulse Imaging Scheme,” *Ultrasonic Imaging*, vol. 17, no. 2, pp. 114–141, 1995.
- [25] “Development Board Raspberry Pi 3 Model B , Computer-Board, BCM2837B0,” *Raspberry Pi 3 Model B | Development Board Raspberry Pi 3 Model B , Computer-Board, BCM2837B0 | RS Components*. [Online]. Available: <https://de.rs-online.com/web/p/entwicklungskits-prozessor-mikrocontroller/1373331>. [Accessed: 27.08.2019, 13:49]
- [26] “40 Way GPIO Rainbow Ribbon Cable - 200mm,” *Pi Supply*. [Online]. Available: <https://uk.pi-supply.com/products/40-way-gpio-rainbow-ribbon-cable-200mm>. [Accessed: 27.08.2019, 14:00]
- [28] “Download Raspbian for Raspberry Pi,” *Raspberry Pi*. [Online]. Available: <https://www.raspberrypi.org/downloads/raspbian/> . [Accessed: 27.08.2019, 14:18]
- [29] “un0rick open hardware,” *un0rick open hardware project* [Online]. Available: <http://un0rick.cc/> . [Accessed: 4 November 2019, 17:32]
- [30] Jonveaux, L., “Echomod Projects,” *Hacking ultrasound with a DIY dev kit*. [Online]. Available: <https://kelu124.gitbooks.io/echomods/content/>. [Accessed: 26 July 2019, 15:28]
- [31]: L. Jonveaux, “un0rick - Open ice40 Ultrasound Imaging Dev Board by kelu124” *Tindie*. [Online]. Available: <https://www.tindie.com/products/kelu124/un0rick-open-ice40-ultrasound-imaging-dev-board/>. [Accessed: 26.August 10:45]
- [32] “Ultimaker 2.” [Online]. Available: <https://www.3dmensionals.de/media/pdf/da/f1/4a/Ultimaker-3-Handbuch.pdf>. [Accessed: 16 August 2019, 17:02]
- [33] Python Software Foundation, “Scientific Python,” *Python.org*. [Online]. Available: <https://www.python.org/success-stories/category/scientific/> . [Accessed: 27.08.2019, 10:23]
- [34] “The SciPy Reference Guide” *SciPy*. [Online]. Available: <https://docs.scipy.org/doc/scipy/reference/> . [Accessed: 5 September 2019, 20:47]
- [35] “NumPy Reference” *NumPy Reference- NumPy v1.18.dev0 Manual*. [Online]. Available: <https://numpy.org/devdocs/reference/index.html> . [Accessed: 10 September 2019]
- [36] *MeVisLab*. [Online]. Available: <https://www.mevislab.de/mevislab/> . [Accessed at: 4.9.2019, 13:44]
- [37] Electrotechnical Commission (IEC): K. Brendel, L. S. Filipczynski, R. Gerstner, C. R. Hill, G. Kossoff, G. Quentin, J. M. Reid, J. Saneyoshi, J. C. Somer, A. A. Tchevnenko, and P. N. T. Wells, “Methods of measuring the performance of ultrasonic pulse-echo diagnostic equipment,” Technical Report. *Ultrasound in Medicine & Biology*, 12-Mar-2004. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0301562977900400> . International [Accessed: 25 October 2019, 11:32]

- [38] U. Sara, Morium Akter, Mohammad Shorif Uddin, " Image Quality Assessment through FSIM, SSIM, MSE and PSNR – A Comparative Study." [Online]. Available: [https://www.researchgate.net/publication/331435447\\_Image\\_Quality\\_Assessment\\_through\\_FSIM\\_SSIM\\_MSE\\_and\\_PSNR-A\\_Comparative\\_Study](https://www.researchgate.net/publication/331435447_Image_Quality_Assessment_through_FSIM_SSIM_MSE_and_PSNR-A_Comparative_Study) . [Accessed: 18 November 2019, 11:05].
- [39] C. F. Dietrich, " Technical Quality Evaluation of diagnostic ultrasound systems - a comprehensive overview of regulations and developments," *EFSUMB - European Course Book*. [Online]. Available: [http://www.kosmos-host.co.uk/efsumb-ecb/coursebook-tqe\\_ch28.pdf](http://www.kosmos-host.co.uk/efsumb-ecb/coursebook-tqe_ch28.pdf) . [Accessed: 25 October 2019, 12:03]
- [40] A. M. Niknejad, "Lecture on Scatter Parameters". *University of Berkeley*, Available: [http://rfic.eecs.berkeley.edu/~niknejad/ee242/pdf/eecs242\\_lect5\\_sparam.pdf](http://rfic.eecs.berkeley.edu/~niknejad/ee242/pdf/eecs242_lect5_sparam.pdf) , [Accessed: 5 November 2019, 14:21]
- [41] M. Ali, D. Magee, U. Dasgupta (Texas Instruments), "Signal Processing Overview of Ultrasound Systems for Medical Imaging". WhitePaper, 2008, [Accessed at: 25.10.2019, 16:54]
- [42] A. P. Dhawan, H. K. Huang, and D.-S. Kim, *Principles and advanced methods in medical imaging and image analysis*. Singapore: World Scientific, 2008.
- [43] John W. Eaton, David Bateman, Søren Hauberg, Rik Wehbring. "GNU Octave version 5.1.0". *manual: a high-level interactive language for numerical computations*, 2019, Available: <https://www.gnu.org/software/octave/doc/v5.1.0/> . [Accessed: 30.10.2019, 9:43]
- [44] H. W. Chesbrough, *Open innovation: the new imperative for creating and profiting from technology*. Boston, MA: Harvard Business School Press, 2011.
- [45] J. M. Pearce, *Quantifying the Value of Open Source Hardware Development* (December 25, 2014). Modern Economy, 2015, 6, 1-11. Available at SSRN: <https://ssrn.com/abstract=3331131>
- [46] R. Tiwari, L. Fischer, K. Kalogerakis, „Frugal Innovation in Germany: A qualitative analysis of socio-economic impacts“. *Working Paper* [Online]. Available: <https://www.econstor.eu/bitstream/10419/156688/1/883183072.pdf> , [Accessed at: 29.10.2019, 14:35]
- [47] C. Y. Baldwin, "Modularity and Organizations." Harvard Business School Working Paper, No. 13-046, November 2012. (To appear in the Elsevier International Encyclopedia of the Social and Behavioral Sciences, 2nd edition; available on request to the author.)
- [48] J. Janjic , P. Kruizinga, P. Van Der Meulen, G. Springeling, F. Mastik, G. Leus, G. van Soest, "Structured ultrasound microscopy". *Applied Physics Letters*, 112(25), 251901, May 2018. Available: <http://adsabs.harvard.edu/abs/2018ApPhL.112y1901J> . [Accessed: 24.10.2019, 20:14]

## 7. Annexe

Code Snippets from MeVisLab Module “LowCostUS”, Tab 3 “USProcessing” are described below to.

### “Flat 2D image” Reconstruction Algorithm

Parameters  $t_4$ , Operating Frequency of transducer ( $f$ ) and down-sampling factor ( $dsampling$ ) are loaded from the **parameters.json** (Tab1) to reconstruct the Flat 2D image array. At this point of the process, the signal is the down-sampled smoothed envelope of the recorded echo ( $smoothedEnvelopeDS$ ). This array’s index represents the number of the acquisition line (first entry:  $smoothedEnvelopeDS[0]$ )

```
offset = int(t4 / 1000 * f / dsampling)
xdim = len(smoothedEnvelopeDS)
ydim = int(offset) + len(smoothedEnvelopeDS[0])
Image = np.zeros([xdim,ydim])

for i in range(xdim):
    for j in range(len(smoothedEnvelopeDS[0])):
        Image[i, offset + j] = list(smoothedEnvelopeDS[i])[j]
```

### “True 2D Image” Reconstruction Algorithm

Parameters concerning the scan angle ( $2*openingAngle$ ), sampling rate and downsampling factor ( $fsampling$ ,  $dsampling$ ), Gaussian-Kernels ( $kernelX$ ,  $kernelY$ ) are loaded into the script from the **parameters.json** which is generated in Tab1 and saved to the specified project folder.

```
#speed of sound in water
vSound = 1480 #in m/s

#distance between two data points on one lineExpl in the image
dist = np.multiply(vSound / (fsampling * 10**6) * 10 ** 3 / 2 , dsampling)
#in mm

trueImageYdim = int(2*np.sin(np.pi*openingAngle/180) * ydim)
trueImageXdim = ydim

TrueImage = np.zeros([trueImageXdim,trueImageYdim])

x0 = 0
y0 = int(trueImageYdim /2)

degpl = 2 * openingAngle / xdim

xdist = int(trueImageXdim * dist)
ydist = 2*np.sin(openingAngle*np.pi/180)*xdist

for i in range(xdim):
    for j in range(ydim):
        x = np.int32(np.cos(np.pi / 180 * (degpl * i - openingAngle) ) * j )
        y = np.int32(np.sin(np.pi / 180 * (degpl * i - openingAngle) ) * j )
        val = Image[i,j]
        TrueImage[x+x0,y0+y] = val

#applying a Gauss filter to the true image
img = ndimage.gaussian_filter(TrueImage, sigma=(kernelX,kernelY))
```