



3 1293 00906 3516

This is to certify that the
dissertation entitled

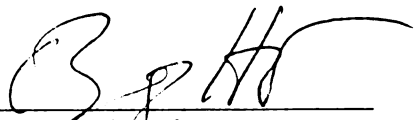
Acoustical Imaging by Broadband Signals

presented by

Nathan Nai-Hsien Wang

has been accepted towards fulfillment
of the requirements for

Ph.D degree in Electrical Engineering


Major professor

Date October 21, 1991



PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.

DATE DUE	DATE DUE	DATE DUE
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____

ACOUSTICAL IMAGING BY BROADBAND SIGNALS

By

Nathan Nai-Hsien Wang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1991

ABSTRACT

ACOUSTICAL IMAGING BY BROADBAND SIGNALS

By

Nathan Nai-Hsien Wang

The conventional method for measuring attenuation and velocity of material generally involves tedious and/or repetitious procedures. Using a narrow acoustic pulse (broad frequency spectrum) for transducer excitation, it is shown that the reflection coefficients of different materials are frequency dependent. Measuring the reflection coefficient as a function of frequency, the velocity-density product and attenuation-density ratio can be obtained with high precision. This technique can be extended to an n-layered structure. A number of different materials are identified based on the dispersion of a narrow acoustic pulse. In addition to the theoretical derivation, the acoustical imaging by broadband signals are constructed. Using hierarchical clustering techniques, the imaging is further characterized. The techniques discussed in this dissertation apply not only to nondestructive evaluation, but also to biological diagnoses.

Copyright by
Nathan Nai-Hsien Wang
1991

To my parents and my wife

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to Dr. Bong Ho and Dr. H. Roland Zapp, my advisors, for their guidance and support. I want to thank Dr. John R. Deller and Dr. Joseph C. Gardiner for their help. Most importantly, I would like to thank my parents, Mrs. and Mr. Chiou-Yueh and Chin-Chi Wang, and my wife Ye-Min Nei. Because of their encouragement and support, I was able to write this dissertation.

TABLE OF CONTENTS

LIST OF TABLES		viii
LIST OF FIGURES		ix
1 INTRODUCTION		1
1.1 Overview		1
1.2 Chapter Descriptions		3
2 MEASUREMENTS AND APPLICATIONS OF ULTRASONIC SIGNALS		4
2.1 Measurements of Ultrasonic Velocity and Attenuation		4
2.1.1 Measurement of Ultrasonic Velocity		5
2.1.2 Measurement of Ultrasonic Attenuation Constant		7
2.2 Nondestructive Evaluation of Materials by Ultrasound		11
2.2.1 Nondestructive Evaluation		11
2.2.2 Nondestructive Evaluation by C-scan		12
2.2.3 Resolution in C-scans		14
2.3 Medical Ultrasonics and Tissue Characterization		20
2.3.1 Acoustical Imaging		20
2.3.2 Tissue Characterization		21
3 BASIC PROPERTIES OF ULTRASOUND		25
3.1 Review of Basic Ultrasound Principles		25
3.2 The Acoustic Plane Wave Equation		26
3.3 Transmission and Reflection Coefficients		31
3.4 Broadband Signal Probing Techniques		34
4 DETERMINATION OF VELOCITY AND ATTENUATION BY BROADBAND SIGNALS		37
4.1 Damping and Attenuation		38

4.2	Derivations under the Assumption that the Attenuation Coefficient Is Proportional to Frequency Squared	42
4.3	Derivations under the Assumption that the Attenuation Coefficient Is Proportional to the l -th Power of Frequency	54
5	EXPERIMENTAL PROCEDURE AND RESULTS	59
5.1	Experimental Procedure	60
5.2	Experimental Results	63
5.3	Two Dimensional Imaging by Using Velocity-Density Product	72
6	ULTRASONIC IMAGES AND TISSUE CHARACTERIZATION BY HIERARCHICAL CLUSTERING TECHNIQUES	79
6.1	Hierarchical Clustering	80
6.2	Material Identification by Ultrasonic Images	87
6.3	Tissue Characterization by Ultrasonic Broadband Signals	92
6.3.1	Ultrasonic Imaging of Tissues	92
6.3.2	Feature Selection	93
6.3.3	Image Enhancement	96
6.3.4	Tissue Characterization by Hierarchical Clustering	100
6.3.5	Data Reduction	102
6.3.6	Image Reconstruction	102
6.3.7	Cluster Validation	104
7	CONCLUSIONS AND DIRECTIONS FOR FUTURE WORK	110
7.1	Conclusions	110
7.2	Directions for Future Work	111
A	PROGRAM LIST - SCANNING AND COLOR DISPLAY	112
	BIBLIOGRAPHY	144

LIST OF TABLES

3.1	Approximate values of ultrasonic velocities of various media	27
5.1	Experimental results of measuring velocity and attenuation by broad-band signals	71
6.1	The values of CPCC by different clustering methods for the 2-layered model	88
6.2	The values of CPCC by different clustering methods for a section of human brain with a hemorrhaged tumor	108

LIST OF FIGURES

3.1	Infinite homogeneous medium with uniform applied pressure	28
3.2	Transmission and reflection at an interface	32
5.1	Computer simulation of the reflected Gaussian pulse from four different materials	61
5.2	The block diagram of the experimental arrangement for reflected mode	62
5.3	The procedure of the sampling program	64
5.4	The procedure for data processing	65
5.5	The incident signal and reflected signals from test materials	67
5.6	The spectra of the incident signal and reflected signals from test materials	68
5.7	The reflection coefficients of test materials within their 3-dB bandwidth	69
5.8	The results of regression with 90% prediction confidence interval . . .	70
5.9	The block diagram of the scanning system	73
5.10	A 2-layered 2-dimensional sample model	75
5.11	Imaging of the 2-layered 2-dimensional sample model using velocity-density product data	76
5.12	A 3-layered 2-dimensional sample model	77
5.13	Imaging of the 3-layered 2-dimensional sample model using velocity-density product data	78
6.1	Clustering result of a simple 2-layered model by single link clustering method with 3 clusters	89
6.2	Clustering result of a simple 2-layered model by complete link clustering method with 2 clusters	89
6.3	Clustering result of a simple 2-layered model by Ward's clustering method with 2 clusters	90
6.4	Clustering result of a simple 2-layered model by UPGMA clustering method with 2 clusters	90
6.5	Comparison of CPCC by the different methods for the 2-layered model	91
6.6	A section of human brain with hemorrhaged tumor	95

6.7	Ultrasonic imaging constructed using the total energy of the reflected signals	95
6.8	Central frequency, peak frequency, and bandwidth of a typical ultrasonic spectrum	97
6.9	Ultrasonic image constructed using the peak frequency of the reflected signals	98
6.10	Ultrasonic image constructed using the central frequency of the reflected signals	98
6.11	Ultrasonic image constructed using the 3-dB bandwidth of the reflected signals	99
6.12	Ultrasonic image constructed using the cross correlation of the incident signal and the reflected signals	99
6.13	The original image obtained from a composite material with a hole in the center	101
6.14	The enhanced image by interpolation	101
6.15	Projection of five features in a two-dimensional space	103
6.16	Clustering result of a section of human brain with a hemorrhaged tumor by single link clustering method with 8 clusters	105
6.17	Clustering result of a section of human brain with a hemorrhaged tumor by complete link clustering method with 4 clusters	105
6.18	Clustering result of a section of human brain with a hemorrhaged tumor by Ward's clustering method with 9 clusters	106
6.19	Clustering result of a section of human brain with a hemorrhaged tumor by UPGMA clustering method with 3 clusters	106
6.20	Comparison of CPCC by the different methods for a section of human brain with a hemorrhaged tumor	109

CHAPTER 1

INTRODUCTION

1.1 Overview

Ultrasonic techniques are becoming increasingly important in medicine [1, 2], non-destructive evaluation [3], and many other applications [4]. Most ultrasonic imaging systems use echo returns from boundaries of different acoustic impedances to show material properties related to boundary variations. It is the purpose of the research reported here to show that the material (or tissue) characteristics can be extracted from the shape of the echo pulse when an extremely narrow pulse excitation is used.

Techniques for the measurement of ultrasonic velocity and attenuation of a medium have existed for a long time, but none permits data over a wide frequency range to be acquired rapidly. Usually a sequence of measurements is made at discrete frequencies over the required range [1, 5]. This is time consuming, and a disadvantage if the system being studied is likely to be changing with time [6]. The broadband video pulse technique provides the best solution for this kind of problem.

The video pulse signal contains a broadband of frequencies. Each frequency component propagates through the material with a different velocity and attenuation depending upon the target and material properties. When all the reflected frequency components are received by the receiving antenna (transducer), the output will be

a video pulse which resembles a spread version of the incident pulse. The reflection of each spectral component is determined by the characteristics of the medium from which it is reflected. If the bandwidth is sufficiently broad, many frequency components can be obtained. Therefore, the frequency dependent velocity and attenuation can be measured by a single pulse.

Although the video pulse technique has been used for a variety of applications in target identification by electromagnetic waves [7, 8], it has not been applied to acoustic probing in spite of the advantage of lower operation frequencies and lower wave attenuation, since the attenuation coefficient is directly related to frequency. In addition to the lower operating frequency, another attractive feature is that the velocity of an acoustic wave (about 1500 *m/sec* in water) is five orders of magnitude lower than that of the electromagnetic wave in free space. Based on the relationship between frequency and wavelength for any propagating wave, $u = f\lambda$, the acoustic wave should have much shorter wavelength which in turn provides superior range resolution.

Since the acoustic wave is a pressure wave, the properties of the acoustic wave are slightly different from the electromagnetic wave. For electromagnetic wave propagation, the dominant characteristics of the medium are the conductivity and the dielectric constant, while for ultrasonic waves, the propagation and reflection depend on the medium density, elasticity, and viscosity.

Theoretically, the characteristics of a medium can be determined if the reflected (or transmitted) and incident pulses are known. Conversely, if the incident pulse and the characteristics of the medium are known, it is possible to calculate the reflected (or transmitted) pulse shape.

Instead of using discrete frequencies for echo measurements, a broad bandwidth acoustic pulse can be utilized to detect the dispersive property of a material for identification purposes. Each frequency component propagates through the material with

a different velocity and attenuation depending upon the properties of the medium. Consequently, the velocity and attenuation constant can be measured by comparing the pulse shape of the echo return to that of the known incident pulse. The acoustical imaging by broadband signals can also be constructed.

1.2 Chapter Descriptions

Chapter 2 reviews the methods to measure velocity and attenuation by ultrasound and applications of ultrasonics. Chapter 3 provides the derivation of acoustic waves and states the basic properties of ultrasound. Chapter 4 provides theoretical derivations of reflections with broadband signals and develops the velocity and attenuation models for multilayer structures. Chapter 5 gives the experimental procedure to verify the derivations developed in Chapter 4. In Chapter 6, a section of human brain with a hemorrhaged tumor is used for experimental measurements. The acoustical images using various features are constructed and using hierarchical clustering methods the image is characterized. Chapter 7 provides a conclusion and some suggestions for future work.

CHAPTER 2

MEASUREMENTS AND APPLICATIONS OF ULTRASONIC SIGNALS

2.1 Measurements of Ultrasonic Velocity and Attenuation

The measurements of ultrasonic velocity and attenuation are very important in ultrasonic applications and many methods for measuring these properties have been developed. Although each method has its own advantages and disadvantages, the choice of optimum method depends on the nature of the particular material and the desired measurement accuracy. In the following sections, we will describe some approaches for measuring ultrasonic velocity and attenuation.

2.1.1 Measurement of Ultrasonic Velocity

Pulse-Superposition Method

The pulse-superposition method allows a very accurate measurement of sound velocity [4, 9, 10]. The radio-frequency (rf) pulses excited by a transducer, are sent to a specimen, from which multiple echoes return. Each succeeding echo is constructively added to the earlier echo of a given pulse by controlling the pulse repetition rate based on the reciprocal of the travel time in the target. The technique is called pulse superposition because the ultrasonic pulses are literally superposed on each other. The high degree of accuracy arises from the fact that the method is capable of precisely measuring the time between cycles. The velocity is twice the thickness of the target times the inverse time separation between pulses.

Sing-Around Method

The sing-around method [4, 10, 11] uses two transducers to measure the velocity of sound by placing one transducer at each end of the test material. One transducer is used as the transmitter the other as the receiver. The receiving ultrasonic pulse transducer generates a trigger signal for the transmitting transducer in order to establish a pulse repetition rate, (PRR). The velocity can therefore be measured by the following equation:

$$v = \frac{length \times PRR}{1 - e \times PRR} \approx length \times PRR \quad (2.1)$$

where PRR is the pulse repetitions rate per second, $length$ is the length of the test material, and e is a correction factor for delays in the transducers, in the coupling between the transducers and the material, and in the electrical components. The measurement of sound velocity by this method is of moderately high accuracy.

Interferometer Method

The interferometer [4] is a continuous-wave (CW) device which has been used for accurately measuring velocity and attenuation of sound in liquids and gases in which standing waves can be sustained. The transducer is fixed at one end of a fluid column with a movable rigid reflector at the other end. The reflector is moved by a micrometer adjustment mechanism. For a fixed frequency as the reflector is moved, the reflected wave generates in phase and out of phase components which add constructively or destructively with the transmitted wave. The half wavelength of the particular sound frequency is determined by the distance the micrometer moves between two successive maxima. The accuracy of the measurement thus depends on the accuracy of the micrometer readings, the parallelism between reflector and transducer surfaces, and the accuracy of the frequency determination.

Resonance Method

The resonance method is similar to the interferometer method. It can be applied to measure the velocity of sound in gases, liquids, or solids. The resonance method uses either one fixed transducer and one fixed reflector, or two transducers located by a known distance apart. By changing the frequency of the ultrasonic wave, the successive resonances can be determined. The difference between two successive resonant frequencies in a nondispersive medium is equal to the fundamental resonant frequency of the medium. From the resonance frequency difference it is possible to determine the sound velocity from the relationship for a single transducer:

$$v = 2l\Delta f \quad (2.2)$$

where l is the distance between the transducer and the reflecting surface and Δf is

the difference between successive resonant frequencies. The accuracy of the resonance method depends on the accuracy of the frequency determination and the measurement of the distance between reflecting surfaces.

Other Methods of Interest

In addition to the above methods, there are several methods in use for measuring the velocity of acoustic waves in special cases [12]-[16]. For examples, McSkimin [17] and Papadakis [10] introduced a method which is particularly good for rf measurements in thin specimens such as rare single crystals and thin sheet metal. Dameron used an inhomogeneous media model, instead of the traditional layered model, to determine the acoustic velocity in tissue [12], and Ophir *et al.* presented a pulse-echo beam-tracking method and transmission method to measure the speed of sound in tissue [18]- [21].

2.1.2 Measurement of Ultrasonic Attenuation Constant

The ultrasonic wave propagating in the z -direction can be defined as:

$$A(t, z) = A_0 e^{-\alpha z} \cos(Kz - \omega t) \quad (2.3)$$

where t is time, A is the amplitude, A_0 is the amplitude at $t = 0$ and $z = 0$, K is the propagation constant, and α is the attenuation constant.

The attenuation constant α can be determined from:

$$\alpha = \frac{\ln(A_1/A_2)}{(z_2 - z_1)} \quad (2.4)$$

where A_1 and A_2 are the amplitudes measured at position z_1 and z_2 at times of maximum amplitude. The unit of the attenuation constant is nepers per unit length.

There are many methods developed to measure the attenuation constant and some will be outlined below.

Spectral-Difference Approach

Kuc introduced the spectral-difference method to measure the attenuation in biological tissue [1], [22]-[24]. If it is assumed that tissue acts like a linear system, the two-way power transfer function, $|H(f)|^2$, is given by:

$$|H(f)|^2 = \frac{P_F(f)}{P_N(f)} \quad (2.5)$$

where $P_F(f)$ and $P_N(f)$ are the spectra of the reflected signals from the far and near surfaces respectively.

Since the acoustic attenuation coefficient of most soft biological tissue has a linear-with-frequency attenuation characteristic, the attenuation coefficient at frequency f , $\alpha(f)$, can be expressed as:

$$\alpha(f) = \beta f \text{ (dB/cm)} \quad (2.6)$$

where β is the slope of the attenuation versus frequency.

The power transfer function of biological tissue can thus be expressed as:

$$|H(f)|^2 = 10^{-\alpha(f)(2D)/10} = 10^{-0.2\beta f D} \quad (2.7)$$

where $2D$ is the additional path length through the tissue traveled by the far surface reflection.

If we use the log-spectra, we have

$$\begin{aligned} 10 \log_{10} |H(f)|^2 &= 10 \log_{10} P_F(f) - 10 \log_{10} P_N(f) \\ &= -2\beta f D \end{aligned} \quad (2.8)$$

Therefore, the value of β can be found by

$$\begin{aligned}\beta &= \frac{-10 \log_{10} |H(f)|^2}{2Df} \\ &= \frac{S_f}{2D} \text{ dB}/(\text{cm} - \text{MHz})\end{aligned}\tag{2.9}$$

where S_f is the slope of the attenuation with respect to frequency of the log-spectral difference.

Spectral-Shift Method

In addition to the spectral-difference method, Kuc introduced the spectral-shift method to measure the attenuation in biological tissue [22]- [24]. The spectral-shift approach requires that the propagating pulse have a Gaussian-shaped spectrum, while the spectral-difference method did not require a specific spectral form.

Using the same notation as in section 2.1.2, the spectrum of the reflected signal from the near surface, P_N , will have a Gaussian shape given by:

$$P_N(f) = C_N e^{-(f-f_N)^2/B^2}\tag{2.10}$$

where C_N is a constant, B is a measure of the pulse bandwidth, and f_N is the centroid of the spectrum, or central frequency.

The two-way power transfer function $|H(f)|^2$ is given by:

$$|H(f)|^2 = e^{-4\beta f D}\tag{2.11}$$

where β is the slope of the attenuation coefficient having units of nepers per centimeter-megahertz. The spectrum from the far surface reflection, $P_F(f)$, is:

$$P_F(f) = C_F e^{-(f-f_F)^2/B^2}\tag{2.12}$$

where C_F is a constant, and

$$f_F = f_N - 2\beta DB^2 \quad (2.13)$$

The spectrum of the far surface reflection still has the same Gaussian form as the spectrum of the near surface reflection, but it has been shifted to a lower center-frequency. Therefore, the value of β can be determined from the frequency shift in the central frequency:

$$\beta = \frac{f_N - f_F}{2DB^2} \text{ Np/(cm-MHz)} \quad (2.14)$$

Since $1 \text{ Np/(cm-MHz)} = 8.686 \text{ dB/(cm-MHz)}$, β can be expressed as follows:

$$\beta = \frac{4.343(f_N - f_F)}{DB^2} \text{ dB/(cm-MHz)} \quad (2.15)$$

Other Methods and Applications

Many papers discuss the measurement of the attenuation constant in different media [25]-[29]. Since the attenuation constant is one of the most important properties of materials (or tissue), many applications use the attenuation constant to classify the medium, such as whether livers are normal or abnormal [30]-[34]. For example, Matsuzawa *et al.* developed a technique for measuring both velocity and attenuation of sound in highly absorptive materials [25], Parker and Waag proposed a method to measure the ultrasonic attenuation from B-scan images in biological applications [31], and Wang *et al.* used the video pulse technique to investigate velocity and attenuation in homogeneous models [35]- [40] as discussed in Chap. 4.

2.2 Nondestructive Evaluation of Materials by Ultrasound

2.2.1 Nondestructive Evaluation

Nondestructive evaluation (NDE) of materials by ultrasound has experienced tremendous growth in recent years, especially for composite materials [4, 41, 42]. The development of reliable test methods for the nondestructive evaluation of composites is a challenging task. In the past ten to fifteen years, a great deal of research has focused on the development of quantitative active ultrasonic testing as well as passive acoustic emission techniques.

Measurements by ultrasound can provide a great deal of information on the properties of materials. Measurement techniques utilizing ultrasonic waves are especially attractive because of the direct connection between the characteristics of the wave propagation and the mechanical properties of a material. The ultrasonic waves can propagate through a material, carrying the information of the samples out without damaging the samples.

In general, a successful ultrasonic materials characterization procedure requires the solution of two problems [42]. The first is related to the reliable detection of the ultrasonic properties, such as amplitude, arrival time, velocity, attenuation, or spectral feature, etc. The second is related to the evaluation of the waveform parameter used to recover the material properties. One can determine the material properties from some particular waveform parameters by establishing a correlation between them.

2.2.2 Nondestructive Evaluation by C-scan

Time-of-Flight C-scans

Conventional ultrasonic C-scans display the amount of energy reflected from some feature in the sample as a function of two dimensions at some fixed depth [3]. Common usage of the C-scan is to locate delamination and other defects, such as porosity, or cracks in materials. Therefore, conventional C-scanning is widely used for quality control during laminate manufacture. However, conventional C-scanning does not provide information on the depthwise distribution of damage. The consequences of having a defect localized between two plies may be quite different from having a uniform distribution of defects throughout the laminate, so that the depth information may be quite important. The depth information can be obtained by recording the position of defects in time by an ultrasonic A-scan and using this time-of-flight information to construct a three dimensional C-scan.

Computer-Automated Measurement and Control

Computer-automated measurement and control (CAMAC) is an instrumentation and interface standard that provides both hardware and software flexibility [43]. By use of the standard CAMAC system a relatively easy interfacing of the motor controllers to the transient recorder for complete computer control is possible. The advantages of the CAMAC system are:

1. Many modules are available, such as analog-to-digital converters and motor controllers.
2. The interface software can be written in a high-level language.
3. New custom applications are easily established by changing modules and writing simple interface routines in high level languages.

C-scan by High Speed Sampling

At present, C-scan ultrasonic imaging is used routinely in examining composite materials [41]. It is well known that there are many attractive features for such a nondestructive evaluation technique. By using the C-scan it is possible to precisely identify voids and defects at various depths inside the material. In practice the following drawbacks limit the application for detecting flaws and defects at various depths inside the material [41]:

1. The range resolution is restricted by the detected pulse width. Improvement can be made by using large amplitude narrow pulses, which however is technically difficult to achieve.
2. It is impossible to obtain a complete impact damage display of a specific fiber layer by the constant-depth scan (C-scan) due to the fact that the fibers at the damage site have been displaced.
3. To detect flaws, all echo returns from the material interior must be processed. This requires a great deal of processing time and thus impedes the effort of rapidly detecting flaws over a large object such as the wing of an aircraft.

By recording the detailed signal waveform and performing some signal processing, the shortcomings mentioned above can be removed. Specifically, the shortcomings can be addressed as discussed below.

Theoretically, the range resolution of an ultrasonic imaging system can approach the operating wavelength. Most commercially available imaging systems offer substantially less resolution than is achievable based on the theoretical limit. By using pulse amplitude detection of echo returns to identify the acoustic boundaries, an uncertainty in range is introduced because of the wide pulse width and the requirement

to exceed a threshold setting in the receiver system. The precision in locating boundaries is further deteriorated by the timing jitter inherent in the transmitted pulse signal.

The difficulty of retrieving the phase information from the rf carrier is due to the high sampling rate required to satisfy the Nyquist criteria. The sampling rate should be 4 or 5 times greater than the signal frequency, or more than 10 MHz for the system utilized in this research. A very high speed real time recording system is necessary to achieve the high sampling rate. A system was designed which involved a very high speed sampling circuit, a memory unit to store the sampled data and the appropriate signal processing software to identify the boundaries. In conjunction with the precise time reference from a fixed spatial reference, the echo returns can be orderly sorted by the software displayed. The unique feature of this signal processing scheme is that the data points can be lined up by layer structure rather than at constant depth from the transducer. Since the retrieved data are stored in groups by layer as well as by the spatial location from the transducer, a three-dimensional contour plot of each layer becomes possible.

2.2.3 Resolution in C-scans

One important topic for C-scan images is the resolution. Resolution refers to the ability to separate echoes from two discontinuities located very close together [4]. To increase the dynamic range usually requires digital rather than analog techniques [2]. The resolution of a C-scan image can be divided into range resolution and lateral resolution. Each of these are considered below.

Range Resolution

The range resolution is the ability to distinguish two different echoes in depth. Since the time axis represents the depth from which the ultrasonic signals were ef-

fect, the problem of range resolution translates to the ability to distinguish signals along the time axis. When a second echo arrives at the receiver before the end of the first these signals will overlap and make it very difficult to distinguish between the two signals. Typically the range resolution is limited by the transmitting pulse width and operating frequency [4]. Pulses with long pulse width will cause poor resolution, while short pulses produce high resolution. In order to develop a high resolution system, a broad bandwidth, low-Q transducer is necessary. Many researchers have tried to increase range resolution by different methods, for example, Lees [44] used the peak ratios of ultrasonic signals to determine the thickness of thin layers. By measuring the peak ratios of the thin layers of known thicknesses, the thickness of unknown layers could be determined by interpolation. This method provides a simple approximation of the layer thicknesses.

In addition to the methods discussed above, another interesting method to increase the range resolution is by deconvolution, as discussed below.

Increasing Range Resolution by Deconvolution

Although the range resolution is related to the ultrasonic wavelength, it is still possible to increase the range resolution beyond the wavelength limitation. The key point is to carefully record the phases of the signals. If it is possible to distinguish the signals of different phases, the resolution is limited only by the identification of phase difference. A powerful tool to assist in phase discrimination and thus to help to increase the range resolution is signal deconvolution.

For a linear time invariant system, deconvolution can be used in many applications. There are many papers which discuss the applications of deconvolution to ultrasonic signals [45]-[49]. Beretsky and Farrell used frequency domain deconvolution to improve ultrasonic imaging [45]. Deconvolution in the frequency domain is straightforward and easy to implement. Assume an incident ultrasonic signal, $x(t)$,

and a reflected signal, $y(t)$ with an impulse response of the test material given by $h(t)$. If the reflection process is linear, the relationship between $x(t)$ and $y(t)$ is:

$$\begin{aligned} y(t) &= x(t) * h(t) \\ &= \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau \end{aligned} \quad (2.16)$$

where $(*)$ is the operation of convolution.

If the Fourier transformations of $x(t)$, $y(t)$, and $h(t)$ are $X(\omega)$, $Y(\omega)$, and $H(\omega)$, respectively, the above relationship between the incident signal and the reflected signal can be written as:

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} \quad (2.17)$$

Therefore, the impulse response of the system can be easily obtained by finding the inverse Fourier transformation of $H(\omega)$:

$$\begin{aligned} h(t) &= \mathcal{F}^{-1} \{H(\omega)\} \\ &= \mathcal{F}^{-1} \left\{ \frac{Y(\omega)}{X(\omega)} \right\} \end{aligned} \quad (2.18)$$

Although this method is straightforward and well known, it has a major difficulty, when the frequency spectrum, $X(\omega)$, is close to zero at some frequencies, which results in the denominator of Eq. 2.17 being close to zero. This will cause substantial error in the calculation of $H(\omega)$. Thus the spectral technique is not usually practical.

Steiner *et al.* introduced a generalized cross-correlator to improve resolution in delay estimation measurements [47]. Yamada presented an on-line deconvolution for the

high resolution ultrasonic pulse-echo measurements using a narrow-band transducer [48]. The on-line deconvolution filter is optimized so as to maximize the resolution capability subject to the constraints of small processing noise.

Silvia summarized many methods of deconvolution [49]. For a discrete-time linear time-invariant system, the deconvolution problem can be solved by a matrix formulation. Assume the sequence of the incident signal is:

$$\begin{aligned} x(i) &\neq 0 \quad i = 0 \dots N - 1 \\ x(i) &= 0 \quad i < 0 \text{ or } i \geq N \end{aligned} \tag{2.19}$$

and the sequence of the impulse response is:

$$\begin{aligned} h(j) &\neq 0 \quad j = 0 \dots M - 1 \\ h(j) &= 0 \quad j < 0 \text{ or } j \geq M \end{aligned} \tag{2.20}$$

and the sequence of the reflected signal is:

$$\begin{aligned} y(k) &\neq 0 \quad k = 0 \dots N + M - 2 \\ y(k) &= 0 \quad k < 0 \text{ or } k \geq N + M - 2 \end{aligned} \tag{2.21}$$

The relationship between $x(n)$ and $y(n)$ can be written as:

$$\begin{aligned} y(n) &= x(n) * h(n) \\ &= \sum_{k=0}^{M-1} h(k)x(n-k) \quad n = 0 \dots N + M - 2 \end{aligned} \tag{2.22}$$

In matrix format, the convolution can be written as:

$$\mathcal{Y} = \mathcal{X}\mathcal{H} \quad (2.23)$$

where

$$\mathcal{Y} = \begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(N + M - 2) \end{bmatrix} \quad (2.24)$$

$$\mathcal{X} = \begin{bmatrix} x(0) & x(-1) & \dots & x(-M + 1) \\ x(1) & x(0) & \dots & x(-M + 2) \\ \vdots & \vdots & & \vdots \\ x(N + M - 2) & x(N + M - 1) & \dots & x(N - 1) \end{bmatrix} \quad (2.25)$$

and

$$\mathcal{H} = \begin{bmatrix} h(0) \\ h(1) \\ \vdots \\ h(M - 1) \end{bmatrix} \quad (2.26)$$

$\mathcal{Y}$ is a $(N + M - 1)$ by 1 matrix, $\mathcal{X}$ is a $(N + M - 1)$ by M matrix, and $\mathcal{H}$ is a M by 1 matrix. Therefore, the impulse response sequence $\mathcal{H}$ can be obtained from:

$$\mathcal{H} = \mathcal{X}^{-1}\mathcal{Y} \quad (2.27)$$

To solve the above equation is usually very difficult because the sizes of $\mathcal{Y}$, $\mathcal{X}$, and $\mathcal{H}$ are usually quit large. Fortunately, many elements in $\mathcal{H}$ can be set to zero by checking the reflected signals $y(n)$. This will greatly reduce the size of matrices and

make the equation solvable.

If there are only K nonzero elements in $\mathcal{H}$, where K is less than or equal to M , then Eq. 2.23 can be rewritten as:

$$\mathcal{Y} = \mathcal{X}_r \mathcal{H}_r \quad (2.28)$$

where $\mathcal{H}_r$ is a reduced matrix from $\mathcal{H}$ obtained by deleting some elements with zero amplitude, and $\mathcal{X}_r$ is a reduced matrix from $\mathcal{X}$ obtained by deleting some columns corresponding to the zeros in $\mathcal{H}$. Thus, $\mathcal{Y}$ is still a $(N + M - 1)$ by 1 matrix, $\mathcal{X}_r$ is a $(N + K - 1)$ by K matrix, and $\mathcal{H}_r$ is a K by 1 matrix.

In Eq. 2.28, there are K unknowns, but we have $(N + M - 1)$ equations. Usually no exact solution can be found since the number of equations is greater than the number of unknowns. However, The solution with the minimum least squares error may be obtained by [50]:

$$\mathcal{H}_r = (\mathcal{X}_r^T \mathcal{X}_r)^{-1} \mathcal{X}_r^T \mathcal{Y} \quad (2.29)$$

where $\mathcal{X}_r^T$ is the transpose of $\mathcal{X}_r$.

Lateral Resolution

Lateral resolution is the ability to distinguish two different objects in the transverse spatial space. Because of the poor focus property of ultrasonic transducers, the lateral resolution is usually low and directly related to the size and frequency of the transducer. To improve resolution, Hundt and Trautenberg used a special digital filter to increase the lateral resolution of B-scan images [51]. The filter is constructed from measured signal amplitudes across the transmitter beam providing a lateral resolution improvement of 50 percent.

Sato, *et al.*, used a homomorphic transform and deconvolution to increase the

sector-scan image [46]. Yokota and Sato developed a super-resolution ultrasonic imaging system by using adaptive focusing [52]. By their method, a set of reflected wave field data from an object was acquired by repeating the transmission and reception for all pairs of transducers on the array. From these data the object-dependent adaptive focusing is performed to illuminate the desired points on the object in order to suppress the signal.

2.3 Medical Ultrasonics and Tissue Characterization

Ultrasound has been widely used in medical applications since the 1970's and the potential for ultrasound utilization is now recognized [2]. Medical ultrasonics is in a period of rapid growth and on the verge of making significant impacts on clinical medicine. Due to fast application specific digital circuit techniques and powerful computers, the applications of ultrasound becomes practical. Many ultrasonic systems and automatic tissue classification systems have recently been developed.

2.3.1 Acoustical Imaging

Ultrasound Volumetric Imaging System

Shattuck *et al.* introduced a parallel processing technique, called explososcan, for high speed ultrasound imaging with linear phased arrays [53]. This technique allows the data acquisition rate to increase by a factor of four by the simultaneous acquisition of four B-mode image lines from each individual broadband transmitted pulse. By using the higher data rate, averaging of the image data to reduce noise is practical.

Smith *et al.* has built a real-time volumetric ultrasound imaging system for medical diagnosis at Duke University [54, 55]. The system uses pulse-echo phased array principles to steer a 2 dimensional array transducer of 289 elements in a pyramidal scan format. By using the explosivescan technique [53], the system can produce 4992 scan lines at a rate of approximately 8 frames/sec by their parallel processing scheme.

Ultrasonic Tomography

In order to obtain quantitative images representing certain tissue properties, ultrasonic computerized tomography has been utilized by Greenleaf [56, 57], Ermert [58], and others.

Ermert and Rohrlein built an ultrasonic reflection mode computerized tomography imaging system by utilizing a conventional B-scanner in a multi-view operation mode [58]. The superposition of several B-scans from different aspect angles of one cross-sectional area and subsequent 2-dimensional processing procedures used in X-ray CT-scanning produced high quality images of the tissue samples.

Another Method of Interest

In addition to the above ultrasonic systems, Richard used time-gain correction methods to enhance ultrasonic B-scan images [59]. This technique achieves signal multiplication by first logarithmically compressing the complete dynamic range of the received echo.

2.3.2 Tissue Characterization

The literature contains many papers on tissue characterization by pattern recognition and segmentation techniques. The objective in this research is to develop automatic detection systems. Usually the choice of classification method is dependent on the object of interest. Feature selection can improve the results impressively.

Tissue Characterization by Bayesian Classifier

Momenan *et al.* applied pattern recognition techniques to ultrasound tissue characterization [60]. They used the Bayesian classifier to determine the feasibility of classifying tissue type as well as determining some properties of the feature spaces. An unsupervised clustering technique, hypothesis, was discussed also. Four features are used in their paper, the average distance between regularly positioned specular scatterers, the ratio of specular to diffuse backscatter intensities, a measure of the variability in the specular component normalized by the diffuse contribution, and the slope of the attenuation coefficient as a function of frequency. The first three features can be obtained directly from the statistics of the squared envelope (intensity data), and the fourth feature can be obtained from the frequency dependent attenuation.

Detection of Tumors by Mammograms

Besides the ultrasonic images, Brzakovic *et al.* presented an automated system for detection and classification of particular types of tumors in digitized mammograms [61]. They used two stages to analyze mammograms, first the system identifies pixel groupings which may correspond to tumors. Second, the detected pixel groupings are subjected to classification. The features used in their work are:

1. area - total number of pixels enclosed in an extracted region.
2. shape descriptor - least mean square difference between the signatures of the tumor model and the extracted region.
3. edge distance variation descriptor - mean square distance between the edge of the tumor model and the locally detected edge from the extracted region.
4. edge intensity variation - local intensity variation along the edges of the extracted region.

They used a classification hierarchy to detect tumors. The first measured feature, which is the easiest feature to measure, can determine whether the samples are non-tumor or tumor. It is only necessary to measure a second feature when the output of the first classification is tumor. The Bayer classifier will be used to perform the second classification. The classification by a third feature is only used if the output of the second feature is abnormal.

Lin *et. al* presents a method for detecting a type of breast tumor, or circumscribed mass from mammograms [62]. It relies on a combination of criteria used by experts, including the shape, brightness contrast, and uniform density of the tumor areas. They used modified median filtering to enhance mammogram images and template matching to detect breast tumors. In the template matching step, suspicious areas are picked by thresholding the cross-correlation values and a percentile method is used to determine a threshold for each film. There are two possible approaches to enhancing mammographic features. One is to increase the contrast of suspicious areas and second is to remove background noise.

Detection of Lung Tumors

Hashimoto *et al.* used a two-feature classification technique for the detection of the edges of x-ray images of lung tumors [63]. The features in this classification technique are the outputs of various forms of gradient and Laplacian operators. They found three two-feature classifiers that are statistically superior to a single feature detector based on the gradient alone in the presence of noise.

Segmentation of Magnetic Resonance Imaging

Magnetic resonance imaging is a powerful tool in medical diagnosis. Bomans *et al.* applied three dimensional segmentation to magnetic resonance imaging [64]. Raya presented a rule-based low-level segmentation system that can automatically

identify the space occupied by different structures of the brain via Magnetic Resonance Imaging [65].

Other Methods of Interest

In addition to the imaging and detection procedures outlined above, other approaches are possible. For example, Trivedi *et al.* used a two-dimensional (gradient, density) feature space for the segmentation of medical images [66], Klingler *et al.* used mathematical morphology to classify the echocardiographic images [67], Michael and Nelson introduced HANDX, a model-based computer vision system for automatic segmentation of bones from digital hand radiographs [68], and Rosenfeld reviews some basic computer vision techniques and speculates about their possible relevance to the modeling of human visual processes [69].

CHAPTER 3

BASIC PROPERTIES OF ULTRASOUND

This chapter provides the theoretical analysis for the ultrasonic wave equations and acoustic broadband signal probing techniques [35, 70, 71]. In section 3.1, some related ultrasound principles are reviewed. The acoustic plane wave equation is derived in section 3.2. In section 3.3, the transmission coefficient and the reflection coefficient are determined from the boundary conditions at interfaces. The properties of the acoustic video pulse will be explained in section 3.4.

3.1 Review of Basic Ultrasound Principles

Ultrasound is the name given to those waves with frequencies above 20 KHz (above the audible range). Ultrasonic waves consist of propagating periodic disturbances in an elastic medium where the particles of the medium vibrate about their equilibrium positions either perpendicular to or parallel to the direction of propagation. Since the longitudinal vibrations are dominant and of lower propagation losses only these variations are considered. The propagation of the resulting motion-strain effects away from the source results in a longitudinal compression wave that transmits mechanical

or acoustic energy away from the source. The vibratory motion of the medium is strongly dependent on the ultrasound frequency and on the state of the medium. Table 3.1 provides the values of velocity of longitudinal sound waves in various materials [4, 71]. Velocity in solids are the highest, those in liquids and biological soft tissues are lower, and those in gases are lowest.

The relationship between frequency and wavelength of a wave is given by

$$\lambda = \frac{u}{f} \quad (3.1)$$

where u is the propagation velocity, f the frequency of the wave, and λ is the wavelength. Since the propagation velocity of ultrasound ($1.5 \times 10^3 \text{ m/sec}$ in water) is much lower than that of an electromagnetic wave ($3 \times 10^8 \text{ m/sec}$ in free space), the wavelength of ultrasound is five orders of magnitude smaller for a given frequency. For a frequency of 5 MHz, the wavelength of ultrasound is approximately 0.3 mm while for an electromagnetic wave it is 60 meters. Because of the much shorter wavelength, the use of ultrasound for detection will give much higher resolution.

3.2 The Acoustic Plane Wave Equation

In this section, we will investigate sound propagation in an infinite homogeneous medium in equilibrium. The coordinate system is shown in Figure 3.1. If a force is applied in the x -direction, it will result in a uniform pressure $p(x_0, t)$ to the $y - z$ plane at a distance x_0 from the origin. Due to this applied force, the particles will experience a displacement in the x -direction. Particles at the location x_0 will also be displaced. If we consider a small differential volume element with incremental length dx and area A in the plane of the applied force, the equilibrium volume, V , of the

Table 3.1. Approximate values of ultrasonic velocities of various media

Material	Mass Density kg/m^3	Longitudinal Velocity m/sec	Acoustic Impedance $\times 10^6 kg/m^2 \cdot sec$
Aluminum	2695	6350	17.1
Copper	8900	4700	42.0
Gold	19300	3240	62.6
Ice	900	3980	3,6
Iron(steel)	7830	5950	46.6
Lead	11400	1960	22.3
Nickel	8800	5600	49.0
Plexiglass	1182	2670	3.17
Silver	10500	3700	39.0
Zinc	7100	4170	29.6
Acetone	790	1174	.93
Alcohol (ethyl)	790	1170	.92
Oil, SAE 20	870	1740	1.50
Water(20°C)	1000	1480	1.5
Air (0°C)	1.293	332	428×10^{-6}
Air (20°C)	1.21	340	411×10^{-6}
Carbon dioxide (16.6°C)	1.935	277	536×10^{-6}
Hydrogen (19.9°C)	0.09	1265	115×10^{-6}
Nitrogen (19.9°C)	1.207	349	421×10^{-6}
Oxygen (19.6°C)	1.38	327	451×10^{-6}

element will be

$$V = A dx \quad (3.2)$$

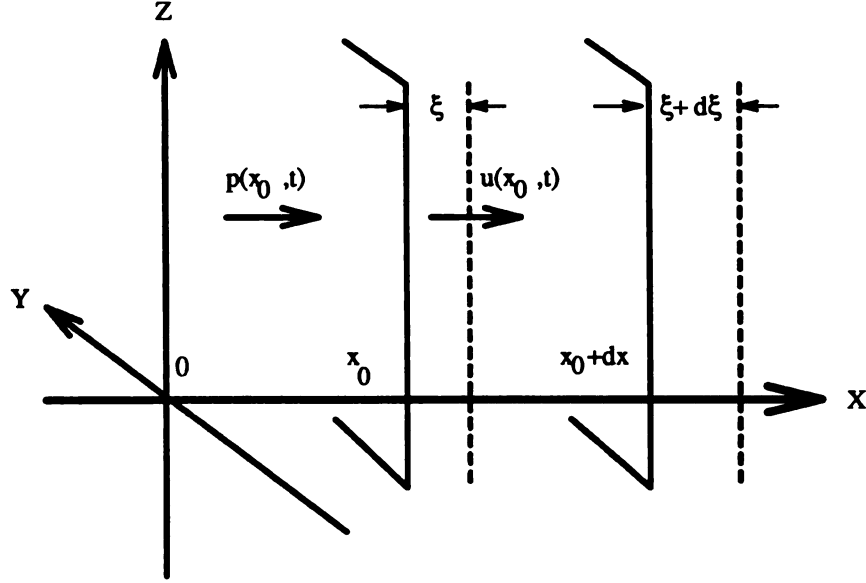


Figure 3.1. Infinite homogeneous medium with uniform applied pressure

As a result of this compressional force the differential volume changes to

$$V' = A(dx + d\xi) = A\left(dx + \frac{\partial \xi}{\partial x} dx\right) \quad (3.3)$$

where $d\xi$ represents a compressional displacement.

Therefore, the change in volume is

$$dV = V' - V = A\left(\frac{\partial \xi}{\partial x} dx\right) \quad (3.4)$$

Conventionally, the strain produced in the volume element is defined as the ratio of

the volume change to the original volume:

$$strain \equiv \frac{dV}{V} = \frac{\partial \xi}{\partial x} \quad (3.5)$$

According to Hooke's law, the ratio of stress to strain in an elastic medium is constant.

This relationship can be expressed as

$$p = -k \frac{\partial \xi}{\partial x} \quad (3.6)$$

where k is the coefficient of elasticity. The negative sign in Eq. 3.6 results from a positive pressure in the x -direction giving a strain in the negative x -direction.

We can define the particle velocity u as the time rate of change of particle displacement. That is,

$$u(x, t) = \frac{\partial \xi}{\partial t} \quad (3.7)$$

The partial derivative of Eq. 3.6, with respect to time, gives:

$$\frac{\partial p}{\partial t} = -k \frac{\partial}{\partial t} \left(\frac{\partial \xi}{\partial x} \right) = -k \frac{\partial u}{\partial x} \quad (3.8)$$

or

$$\frac{\partial u}{\partial x} = \frac{-1}{k} \frac{\partial p}{\partial t} \quad (3.9)$$

If the applied pressure varies with time, the pressure magnitude will be a function of distance, x , as well as of time, t . From Newton's second law of motion, the acceleration of a material element resulting from an applied pressure is:

$$\frac{\partial p}{\partial x} = -\rho \frac{\partial u}{\partial t} \quad (3.10)$$

The negative sign in Eq. 3.10 results from a net acceleration to the right for a

negative spatial pressure gradient. In Eq. 3.10 ρ is the medium density.

Equations 3.9 and Eq. 3.10 are the coupled pressure particle velocity. By decoupling these equations, it is possible to obtain the acoustic plane wave equations

$$\frac{\partial^2 p}{\partial t^2} = \frac{k}{\rho} \frac{\partial^2 p}{\partial x^2} \quad (3.11)$$

and,

$$\frac{\partial^2 u}{\partial t^2} = \frac{k}{\rho} \frac{\partial^2 u}{\partial x^2} \quad (3.12)$$

The general solution for pressure and particle velocity are

$$p = p(0)e^{j(\omega t - Kx)} \quad (3.13)$$

$$u = u(0)e^{j(\omega t - Kx)} \quad (3.14)$$

where K is the wave propagation constant given by

$$K = \omega \sqrt{\rho/k} \quad (3.15)$$

The wave number K is in general a complex quantity. It consists of the phase constant β and the attenuation constant α ,

$$K = \beta - j\alpha \quad (3.16)$$

From Eq. 3.10, 3.13, and 3.14, the relationship between pressure and particle velocity is

$$p = \frac{\omega \rho}{K} u \quad (3.17)$$

The characteristic acoustic impedance is defined as the ratio of the pressure to particle

speed. From Eq. 3.15, the acoustic impedance Z can be expressed as

$$Z \equiv \frac{p}{u} = \frac{\omega \rho}{K} \quad (3.18)$$

For a lossy medium, the acoustic impedance is a complex quantity. For a lossless medium the attenuation constant α is zero, and the wave number reduces to $K = \beta$, which gives a real acoustic impedance.

Under the lossless assumption, the phase velocity v_p is

$$v_p = \frac{\omega}{\beta} = \frac{\omega}{K} \quad (3.19)$$

Therefore, the acoustic impedance for a lossless medium can be written as

$$Z = \rho v_p \quad (3.20)$$

3.3 Transmission and Reflection Coefficients

When an acoustic plane wave is incident to a boundary between two different media, it will be partially reflected. The ratio of the characteristic impedances of the two media determines the magnitude of the reflection coefficient and transmission coefficient. Assume there is an acoustic plane wave propagating from medium 1 to medium 2, as shown in Figure 3.2. Snell's law states that the angles of incidence and reflection are equal when the wavelength of the wave is small compared to the dimensions of the reflector. That is,

$$\theta_i = \theta_r \quad (3.21)$$

From the relationship between θ_i and θ_t , the angle of transmission, can be obtained from:

$$\frac{\sin \theta_i}{\sin \theta_t} = \frac{u_1}{u_2} \quad (3.22)$$

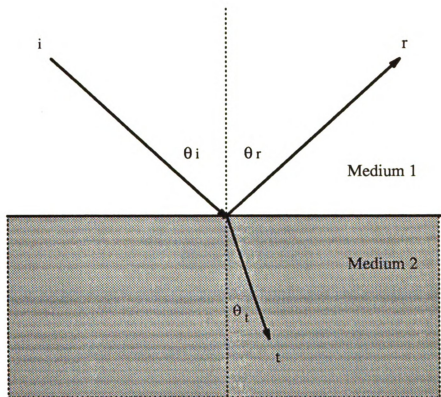


Figure 3.2. Transmission and reflection at an interface

In the equilibrium state, the pressure on both sides of the boundary remains the same in order to maintain a stationary boundary. This gives

$$p_i + p_r = p_t \quad (3.23)$$

Furthermore, the normal component of the particle velocity must be the same on

both sides of the boundary or else the two media will not remain in contact. Thus,

$$u_i \cos \theta_i - u_r \cos \theta_r = u_t \cos \theta_t \quad (3.24)$$

From Eqs. 3.17 and 3.24, we have

$$\frac{p_i K_1 \cos \theta_i}{\rho_1} - \frac{p_r K_1 \cos \theta_r}{\rho_1} = \frac{p_t K_2 \cos \theta_t}{\rho_2} \quad (3.25)$$

Solving Eqs. 3.23 and 3.25, the pressure ratios are

$$\frac{p_r}{p_i} = \frac{\frac{K_1}{\rho_1} \cos \theta_i - \frac{K_2}{\rho_2} \cos \theta_t}{\frac{K_1}{\rho_1} \cos \theta_i + \frac{K_2}{\rho_2} \cos \theta_t} \quad (3.26)$$

and

$$\frac{p_t}{p_i} = \frac{\frac{2K_1}{\rho_1} \cos \theta_i}{\frac{K_1}{\rho_1} \cos \theta_i + \frac{K_2}{\rho_2} \cos \theta_t} \quad (3.27)$$

For normal incidence $\theta_i = \theta_t = 0$, the pressure amplitude ratios reduce to

$$\text{Reflection coefficient, } R = \frac{p_r}{p_i} = \frac{\frac{K_1}{\rho_1} - \frac{K_2}{\rho_2}}{\frac{K_1}{\rho_1} + \frac{K_2}{\rho_2}} \quad (3.28)$$

$$\text{Transmission coefficient, } T = \frac{p_t}{p_i} = \frac{\frac{2K_1}{\rho_1}}{\frac{K_1}{\rho_1} + \frac{K_2}{\rho_2}} \quad (3.29)$$

Using the acoustic impedance definition $Z = \omega \rho / K$, the reflection coefficient and transmission coefficient can be reduced to:

$$R = \frac{Z_2 - Z_1}{Z_2 + Z_1} \quad (3.30)$$

$$T = \frac{2Z_2}{Z_2 + Z_1} \quad (3.31)$$

Up to this point, the reflection and transmission coefficients were expressed in the

time domain. Using the Fourier Transform the reflection and transmission coefficients can be expressed in the frequency domain as well.

The Fourier Transform of $p(x, t)$ is defined as

$$P(x, \omega) = \int_{-\infty}^{\infty} p(x, t) e^{-j\omega t} dt \quad (3.32)$$

Because the reflection coefficient R and the transmission coefficient T are independent of time, the R and T can be written as

$$R = \frac{P_r(0, \omega)}{P_i(0, \omega)} = \frac{\frac{K_1}{\rho_1} - \frac{K_2}{\rho_2}}{\frac{K_1}{\rho_1} + \frac{K_2}{\rho_2}} = \frac{Z_2 - Z_1}{Z_2 + Z_1} \quad (3.33)$$

$$T = \frac{P_t(0, \omega)}{P_i(0, \omega)} = \frac{\frac{2K_1}{\rho_1}}{\frac{K_1}{\rho_1} + \frac{K_2}{\rho_2}} = \frac{2Z_2}{Z_2 + Z_1} \quad (3.34)$$

3.4 Broadband Signal Probing Techniques

Reflection coefficients and transmission coefficients are crucial factors in determining wave propagation in media with different material properties. When the incident wave is a narrow band signal, the reflection coefficient and the transmission coefficient are usually constant. This is not true for a broadband signal. The basic principle of using video pulses is to exploit the broadband signal characteristics to detect media variations. The broadband signal from the transducer is dispersed by the frequency dependent reflection and transmission at the interfaces of different acoustic impedances of different media. The energy of the reflected wave or of the transmitted wave depends on the frequency dependent reflection coefficient or the frequency dependent transmission coefficient.

A necessary requirement for using video pulses is that the waveform of the incident wave in both time and frequency must be well known. When the video pulse

impacts on the interface between the first and second media, the reflected wave contains all frequency components, each modified by its own frequency dependent reflection coefficient. The reflected signal spectrum contains the information necessary to characterize the unknown media.

The Fourier spectrum of the incident pressure wave is given by:

$$P_i(x, \omega) = \int_{-\infty}^{\infty} p_i(x, t) e^{-j\omega t} dt \quad (3.35)$$

If x is set to zero at the interface of two media, the pressure in the frequency domain at the interface is:

$$P_i(0, \omega) = \mathcal{F}\{p(0, t)\} \quad (3.36)$$

where $\mathcal{F}\{\cdot\} = \int_{-\infty}^{\infty} \cdot e^{-j\omega t} dt$ is the Fourier transformation.

In section 3.3, it was shown that the reflection coefficient is independent of time, so that the Fourier Transform of the reflected wave at the interface in the frequency domain is given by:

$$P_r(0, \omega) = P_i(0, \omega)R \quad (3.37)$$

and the reflected pressure wave at the interface in the time domain is

$$p_r(0, t) = \mathcal{F}^{-1}\{P_r(0, \omega)\} = \mathcal{F}^{-1}\{P_i(0, \omega)R\} \quad (3.38)$$

where $\mathcal{F}^{-1}\{\cdot\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \cdot e^{j\omega t} d\omega$ is the inverse Fourier transform and R is the time independent reflection coefficient.

A detected signal a distance d from the interface has the form:

$$p_r(d, t) = p_r(0, t) e^{-jK_1 d} \quad (3.39)$$

The signal to be detected is thus given by:

$$\begin{aligned}
 p_r(x, t) &= \text{Real} \left\{ \mathcal{F}^{-1} \{ P_i(0, \omega) R \} e^{-jK_1 d} \right\} \\
 &= \text{Real} \left\{ \mathcal{F}^{-1} \{ \mathcal{F} \{ p_i(0, t) \} R \} e^{-jK_1 d} \right\}
 \end{aligned} \tag{3.40}$$

The real part of the pressure is taken since only this signal can be detected experimentally.

CHAPTER 4

DETERMINATION OF VELOCITY AND ATTENUATION BY BROADBAND SIGNALS

In this chapter the acoustic plane wave equations including attenuation is derived in 4.1. Based on the wave equations, the measurement of velocity-density product and attenuation-density ratio by broadband signals will be discussed from two different conditions. The first assumes the attenuation coefficient is proportional to the frequency squared. The theoretical derivation is shown in section 4.2. In order to generalize the frequency dependent attenuation, the attenuation coefficient is assumed proportional to the l -th power of frequency, where l is a real number between 1 and 2. The details of this condition are discussed in section 4.3

4.1 Damping and Attenuation

As discussed in section 3.2, the acoustic wave equation Eqs. 3.11 and 3.12 was derived for the condition that the wave propagating through the medium suffers no energy loss, that is, the acoustic wave propagates in the medium without attenuation. Ideal materials of this kind do not exist, although weakly damped materials are often approximated as ideal. Elastic damping usually depends on temperature, frequency, and the type of vibration. At room temperature, acoustic losses in many materials may be adequately described by adding a viscous damping term. In this section, the damping force is used to obtain the attenuation constant α and phase constant β .

In an ideal lossless medium, Hooke's law describes the linear relationship between pressure and strain:

$$p_{ideal} = -cS \quad (4.1)$$

where c is a proportional constant, called elastic stiffness constant, while S is strain, as defined in Eq. 3.5. The damping force in material can be put in the form:

$$p_{damping} = -\eta \frac{\partial S}{\partial t} \quad (4.2)$$

where η is the material viscosity.

The total pressure becomes

$$p = p_{ideal} + p_{damping} = -(kS + \eta \frac{\partial S}{\partial t}) \quad (4.3)$$

For a one-dimensional problem, the strain is

$$S = \frac{\partial \xi}{\partial x} \quad (4.4)$$

From Eqs. 3.7 and 4.4 we get

$$\frac{\partial u}{\partial x} = \frac{\partial^2 \xi}{\partial x \partial t} = \frac{\partial S}{\partial t} \quad (4.5)$$

Taking a partial derivative with respect to time in Eq.4.3, and using Eqs. 4.4 and 4.5 we obtain:

$$\frac{\partial p}{\partial t} = - \left(c \frac{\partial u}{\partial x} + \eta \frac{\partial^2 u}{\partial x \partial t} \right) \quad (4.6)$$

From Eq. 3.10, the spatial derivative of pressure and the time derivative of particle velocity are related by

$$\frac{\partial p}{\partial x} = -\rho \frac{\partial u}{\partial t} \quad (4.7)$$

Following the same procedure used in section 3.2 to decouple Eqs. 4.6 and 4.7, the acoustic plane wave equations including attenuation become:

$$\frac{\partial^2 p}{\partial t^2} = \frac{c}{\rho} \frac{\partial^2 p}{\partial x^2} + \frac{\eta}{\rho} \frac{\partial^3 p}{\partial x^2 \partial t} \quad (4.8)$$

and,

$$\frac{\partial^2 u}{\partial t^2} = \frac{c}{\rho} \frac{\partial^2 u}{\partial x^2} + \frac{\eta}{\rho} \frac{\partial^3 u}{\partial x^2 \partial t} \quad (4.9)$$

The general solutions for pressure and velocity are

$$p = p(0)e^{j(\omega t - Kx)} \quad (4.10)$$

$$u = u(0)e^{j(\omega t - Kx)} \quad (4.11)$$

where

$$K = \beta - j\alpha \quad (4.12)$$

Using these solutions, the following equations are obtained:

$$\rho\omega^2 p = cK^2 p + j\eta K^2 \omega p \quad (4.13)$$

$$\rho\omega^2 = cK^2 + j\eta K^2 \omega \quad (4.14)$$

With $K^2 = \beta^2 - 2\alpha\beta j - \alpha^2$, and both α and β real, the following equations are obtained by equating the real and imaginary parts of Eq. 4.14:

$$c(\beta^2 - \alpha^2) + 2\alpha\beta\eta\omega = \rho\omega^2 \quad (4.15)$$

$$\eta\omega(\beta^2 - \alpha^2) - 2\alpha\beta c = 0 \quad (4.16)$$

Solving Eqs. 4.15 and 4.16 for real α and β , gives:

$$\alpha^2 = \frac{1}{2} \frac{c\rho\omega^2}{(\eta^2\omega^2 + c^2)} [\sqrt{1 + (\eta\omega/c)^2} - 1] \quad (4.17)$$

and

$$\beta^2 = \frac{1}{2} \frac{c\rho\omega^2}{(\eta^2\omega^2 + c^2)} [\sqrt{1 + (\eta\omega/c)^2} + 1] \quad (4.18)$$

Finally,

$$\alpha = \omega \left\{ \frac{1}{2} \frac{c\rho}{(\eta^2\omega^2 + c^2)} [\sqrt{1 + (\eta\omega/c)^2} - 1] \right\}^{1/2} \quad (4.19)$$

$$\beta = \omega \left\{ \frac{1}{2} \frac{c\rho}{(\eta^2\omega^2 + c^2)} [\sqrt{1 + (\eta\omega/c)^2} + 1] \right\}^{1/2} \quad (4.20)$$

The above equations give the general solution for the attenuation constant α and phase constant β . Eqs. 4.19 and 4.20 indicate that both the attenuation constant α and the phase constant β are functions of frequency. We can express c and η in terms

of α and β as:

$$c = \frac{\rho\omega^2(\beta^2 - \alpha^2)}{(\beta^2 + \alpha^2)^2} \quad (4.21)$$

and,

$$\eta = \frac{2\rho\omega\alpha\beta}{(\beta^2 + \alpha^2)^2} \quad (4.22)$$

For a given frequency, the attenuation constant α is proportional to the square root of the medium density. As the viscosity constant increases, the attenuation constant also increases. If the elasticity stiffness constant increases, the attenuation constant decreases.

The acoustic impedance of material is given by [70, 72]:

$$Z = \omega\rho/K \quad (4.23)$$

where $K = \beta - j\alpha$, is the complex wave propagation constant. As a result, for lossy media, the acoustic impedance is a complex quantity as well. At the boundary between media i and $i + 1$, for normal incidence, the reflection coefficient becomes:

$$\begin{aligned} R_i &= \frac{Z_{i+1} - Z_i}{Z_{i+1} + Z_i} \\ &= \frac{(K_i/\rho_i) - (K_{i+1}/\rho_{i+1})}{(K_i/\rho_i) + (K_{i+1}/\rho_{i+1})} \\ &= \frac{(\beta_i/\rho_i - \beta_{i+1}/\rho_{i+1}) - j(\alpha_i/\rho_i - \alpha_{i+1}/\rho_{i+1})}{(\beta_i/\rho_i + \beta_{i+1}/\rho_{i+1}) - j(\alpha_i/\rho_i + \alpha_{i+1}/\rho_{i+1})} \end{aligned} \quad (4.24)$$

where $j = \sqrt{-1}$. Thus,

$$|R_i|^2 = \frac{(\beta_i/\rho_i - \beta_{i+1}/\rho_{i+1})^2(\alpha_i/\rho_i - \alpha_{i+1}/\rho_{i+1})^2}{(\beta_i/\rho_i + \beta_{i+1}/\rho_{i+1})^2(\alpha_i/\rho_i + \alpha_{i+1}/\rho_{i+1})^2} \quad (4.25)$$

From above equation, the reflection coefficient is a function of the attenuation constant, the phase constant, and the material density. Therefore, the reflection coefficient will contain information of α , β , and ρ .

4.2 Derivations under the Assumption that the Attenuation Coefficient Is Proportional to Frequency Squared

In this section measurements of the velocity-density product and attenuation-density ratio of materials for isotropic, homogeneous, linear, lossy media are developed. For typical materials and frequencies of operation (where $(\frac{\eta\omega}{c})^2 \ll 1$), the attenuation and phase constants can be simplified from Eqs. 4.19 and 4.20 to:

$$\begin{aligned} \alpha &\approx \frac{\eta\omega^2}{2c}\sqrt{\rho/c}[1 - (\eta\omega/c)^2]^{1/2} \\ &\approx \frac{\eta\omega^2}{2c}\sqrt{\rho/c}[1 - \frac{1}{2}(\frac{\eta\omega}{c})^2] \\ &\approx \frac{\eta\omega^2}{2c}\sqrt{\rho/c} \\ &\equiv a\omega^2 \end{aligned} \quad (4.26)$$

$$\begin{aligned}
\beta &\approx \omega\sqrt{\rho/c}[1 - \frac{3}{4}(\frac{\eta\omega}{c})^2]^{1/2} \\
&\approx \omega\sqrt{\rho/c}[1 - \frac{3}{8}(\frac{\eta\omega}{c})^2] \\
&\approx \omega\sqrt{\rho/c} \\
&\equiv b\omega
\end{aligned} \tag{4.27}$$

where a and b are constants for given materials,

$$a \equiv \frac{\eta}{2c}\sqrt{\rho/c} \tag{4.28}$$

and

$$b \equiv \sqrt{\rho/c} \tag{4.29}$$

From Eqs. 4.26 and 4.27, when the frequency of the acoustic wave is low $(\frac{\eta\omega}{c})^2 \ll 1$, the attenuation constant is proportion to ω^2 , and the phase constant is proportion to ω . For very low frequencies, the attenuation approaches zero. For higher frequencies the attenuation increases rapidly. The longitudinal acoustic wave velocity in isotropic media becomes

$$v = \frac{\omega}{\beta} \approx \frac{1}{b} \tag{4.30}$$

From the approximations of α and β in Eqs. 4.26 and 4.30, it is clear that the attenuation coefficient is proportional to frequency squared and the longitudinal acoustic wave velocity is constant. Therefore, the following expression can be obtained from 4.25:

$$|R_i(\omega)|^2 = \frac{[(b_i/\rho_i) - (b_{i+1}/\rho_{i+1})]^2 + [(a_i/\rho_i) - (a_{i+1}/\rho_{i+1})]^2\omega^2}{[(b_i/\rho_i) + (b_{i+1}/\rho_{i+1})]^2 + [(a_i/\rho_i) + (a_{i+1}/\rho_{i+1})]^2\omega^2} \tag{4.31}$$

For various frequency components of the echo return, a set of simultaneous equations in the form of Eq. 4.31 can be established. In order to demonstrate the procedure for identifying unknown material, the following cases are considered:

Case 1: The i th medium is lossless ($\alpha_i = 0$)

For simplicity, consider water as the first medium, which is most often used in acoustic imaging setups. Water is an excellent acoustic transmission medium with practically no attenuation in a finite length for frequencies well into the GHz region. Setting the water attenuation coefficient $\alpha_i = 0$, Eq. 4.31 reduces to:

$$|R_i(\omega)|^2 \equiv \frac{A + B\omega^2 - 1}{A + B\omega^2 + 1} \quad (4.32)$$

where A and B are frequency independent real coefficients:

$$A \equiv \frac{(b_i/\rho_i)^2 + (b_{i+1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.33)$$

$$B \equiv \frac{(a_{i+1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.34)$$

Assume the measurement error of $|R_i(\omega_k)|^2$ is ϵ_k , that is,

$$|R_i(\omega_k)|^2 = \frac{A + B\omega_k^2 - 1}{A + B\omega_k^2 + 1} + \epsilon_k \quad (4.35)$$

Equation 4.35 must now be solved for A and B by regression. Unfortunately, this is a nonlinear model for which it is difficult to get a closed form solution. In order to linearize the equation, assume the measurement error ϵ is small, that is, $\epsilon_k^2 \approx 0$.

Thus Eq. 4.35 can be rearranged as follows:

$$(A + B\omega_k^2)[1 - |R_i(\omega_k)|^2 + \epsilon_k]^2 = [1 - |R_i(\omega_k)|^2 - \epsilon_k][1 + |R_i(\omega_k)|^2 + \epsilon_k] \quad (4.36)$$

$$\begin{aligned} (A + B\omega_k^2)[1 - |R_i(\omega_k)|^2]^2 + 2\epsilon_k(A + B\omega_k^2)[1 - |R_i(\omega_k)|^2] + (A + B\omega_k^2)\epsilon_k^2 \\ = [1 - |R_i(\omega_k)|^2][1 + |R_i(\omega_k)|^2] + \epsilon_k|R_i(\omega_k)|^2 - \epsilon_k^2 \end{aligned} \quad (4.37)$$

For $\epsilon_k^2 \approx 0$, this expression can be reduced to:

$$\begin{aligned} [A + B\omega_k^2][1 - |R_i(\omega_k)|^2]^2 - [1 - |R_i(\omega_k)|^2][1 + |R_i(\omega_k)|^2] \\ \approx 2\epsilon_k|R_i(\omega_k)|^2 - 2\epsilon_k(A + B\omega_k^2)[1 - |R_i(\omega_k)|^2] \end{aligned} \quad (4.38)$$

The second term in the right hand side of Eq. 4.38 can be further simplified to:

$$\begin{aligned} \epsilon_k (A + B\omega_k^2)[1 - |R_i(\omega_k)|^2] \\ = \frac{\epsilon_k(1 - |R_i(\omega_k)|^2)(1 + |R_i(\omega_k)|^2 - \epsilon_k)}{1 - |R_i(\omega_k)|^2 + \epsilon_k} \\ = \epsilon_k(1 + |R_i(\omega_k)|^2) - \epsilon_k^2 \left(\frac{1 + |R_i(\omega_k)|^2 - \epsilon_k}{1 - |R_i(\omega_k)|^2 + \epsilon_k} \right) \\ \approx \epsilon_k(1 + |R_i(\omega_k)|^2) \end{aligned} \quad (4.39)$$

Therefore, a linear model can be constructed as follows:

$$\begin{aligned}
& [A + B\omega_k^2][1 - |R_i(\omega_k)|^2]^2 - [1 - |R_i(\omega_k)|^2][1 + |R_i(\omega_k)|^2] \\
& \approx 2\epsilon_k |R_i(\omega_k)|^2 - 2\epsilon_k [1 + |R_i(\omega_k)|^2] \\
& = -2\epsilon_k
\end{aligned} \tag{4.40}$$

Therefore, equation 4.36 reduces to the linearized mode:

$$[A + B\omega_k^2][1 - |R_i(\omega_k)|^2]^2 \approx [1 - |R_i(\omega_k)|^2][1 + |R_i(\omega_k)|^2] - 2\epsilon_k \tag{4.41}$$

Assuming the random error 2ϵ is normally distribution with standard deviation σ , the solutions of A and B , with the minimum sum-of-the-squares error of n measurement data points, $\hat{A}$ and $\hat{B}$, are [50, 73]:

$$\begin{bmatrix} \hat{A} \\ \hat{B} \end{bmatrix} \approx (X^T X)^{-1} X^T Y \tag{4.42}$$

where X is a n by 2 matrix and Y is a n by 1 matrix as follows:

$$X = \begin{bmatrix} (1 - |R_i(\omega_1)|^2)^2 & (1 - |R_i(\omega_1)|^2)^2 \omega_1^2 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ (1 - |R_i(\omega_n)|^2)^2 & (1 - |R_i(\omega_n)|^2)^2 \omega_n^2 \end{bmatrix} \tag{4.43}$$

$$Y = \begin{bmatrix} (1 - |R_i(\omega_1)|^2)(1 + |R_i(\omega_1)|^2) \\ \cdot \\ \cdot \\ \cdot \\ (1 - |R_i(\omega_n)|^2)(1 + |R_i(\omega_n)|^2) \end{bmatrix} \quad (4.44)$$

and X^T is the transpose of X .

In general, for a linear regression model, if the random error ϵ is normally distributed, the estimated values $\hat{A}$ and $\hat{B}$ will be unbiased, normally distributed estimators of A and B respectively [50]. The variances of $\hat{A}$ and $\hat{B}$ will be:

$$\text{Var}(\hat{A}) = c_A \sigma^2 \quad (4.45)$$

and

$$\text{Var}(\hat{B}) = c_B \sigma^2 \quad (4.46)$$

where c_A and c_B are the diagonal elements in $(X^T X)^{-1}$, or

$$\begin{bmatrix} c_A & c_{AB} \\ c_{BA} & c_B \end{bmatrix} \equiv (X^T X)^{-1} \quad (4.47)$$

We must either know σ in Eqs. 4.45 and 4.46 or possess a good estimate of σ so that the variances of $\hat{A}$ and $\hat{B}$ can be estimated. In general, the value σ is unknown so that the estimated standard deviation, S , should be calculated as follows [50, 73]:

$$S \equiv \frac{Y^T Y - \begin{bmatrix} \hat{A} & \hat{B} \end{bmatrix} X^T Y}{n - 2} \quad (4.48)$$

In order to find the 90 % confidence interval of $\hat{A}$, a hypothesis H_0 is constructed

as follows:

$$H_0 : A = A_0 \quad (4.49)$$

where A_0 is a specified value of A . The quantity T_A , which is defined as:

$$T_A = \frac{\hat{A} - A_0}{S\sqrt{c_A}} \quad (4.50)$$

will have a Student's t distribution with $n - 2$ degrees of freedom. Therefore, the 90% confidence interval for $\hat{A}$ is [50, 73]:

$$\hat{A} \pm t_{0.05, n-2} S\sqrt{c_A} \quad (4.51)$$

where $t_{0.05, n-2}$ is a t distribution with $(n - 2)$ degrees of freedom and 0.05 refers to the 90% confidence and $c_A S^2$ is the estimated variance of $\hat{A}$.

By the same method for finding the confidence interval of $\hat{A}$, the 90% confidence interval for $\hat{B}$ is:

$$\hat{B} \pm t_{0.05, n-2} S\sqrt{c_B} \quad (4.52)$$

where $c_B S^2$ is the estimated variance of $\hat{B}$.

With the properties of the i th medium known, the material properties in the $(i+1)$ th medium, such as $(v_{i+1}\rho_{i+1})$ and $(\alpha_{i+1}/\rho_{i+1})$, can be determined. The velocity-density product in the second medium can be obtained, using Eqs. 4.30 and 4.33, as follows:

$$v_{i+1}\rho_{i+1} = \rho_{i+1}b_{i+1} = \frac{v_i\rho_i}{[\hat{A} \pm \sqrt{\hat{A}^2 - 1}]} \quad (4.53)$$

The attenuation-density ratio, using Eqs. 4.26, 4.33, and 4.34, becomes

$$\frac{\alpha_{i+1}}{\rho_{i+1}} = \left(\frac{1}{v_i\rho_i}\right) \left(\sqrt{2[\hat{A} \pm \sqrt{\hat{A}^2 - 1}]} \hat{B}\right) \omega^2 \quad (4.54)$$

The $\pm$ sign in Eqs. 4.53 and 4.54 can be determined by comparing the phases of the incident signal and the reflected signal. If they are in phase, which implies that the impedance of the second medium is greater than that of the first medium, the “minus” sign is used. Otherwise, the “plus” sign should be chosen.

After $\hat{A}$ and $\hat{B}$ are evaluated, the velocity-density product as well as the attenuation-density ratio can be obtained from Eqs. 4.53 and 4.54.

Case 2: The i th medium is lossy ($\alpha_i \neq 0$)

In general, the attenuation constant of the $(i+1)$ th medium is not equal to zero. For such cases the procedure for obtaining velocity-density products and attenuation-density ratios becomes more complicated. If (b_i/ρ_i) , (a_i/ρ_i) , and $|R_i(\omega)|$ are known, the following equation can be constructed from Eq. 4.31:

$$|R_i(\omega)|^2 \equiv \frac{C + (D - H)\omega^2 - 1}{C + (D + H)\omega^2 + 1} \quad (4.55)$$

where C , D , and H are the frequency independent real coefficients:

$$C \equiv \frac{(b_i/\rho_i)^2 + (b_{i+1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.56)$$

$$D \equiv \frac{(a_i/\rho_i)^2 + (a_{i+1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.57)$$

$$H \equiv \frac{a_i a_{i+1}}{b_i b_{i+1}} \quad (4.58)$$

Since there are only two unknown quantities, (b_{i+1}/ρ_{i+1}) and (a_{i+1}/ρ_{i+1}) , in Eq. 4.55, we can choose C and D as independent variables and H as a dependent variable of C and D , expressed as:

$$H = \frac{a_i \sqrt{2D(C \pm \sqrt{C^2 - 1})(b_i/\rho_i)^2 - a_i^2}}{b_i(C \pm \sqrt{C^2 - 1})} \quad (4.59)$$

Assume the measurement error of $|R_i(\omega_k)|^2$ in Eq. 4.55 is ε_k , and ε has a normal distribution with zero mean. Therefore,

$$|R_i(\omega_k)|^2 = \frac{C + (D - H)\omega_k^2 - 1}{C + (D + H)\omega_k^2 + 1} + \varepsilon_k \quad (4.60)$$

The sum of the square error, $E_i(C, D)$, is:

$$\begin{aligned} E_i(C, D) &\equiv \sum_{k=1}^n \varepsilon_k^2 \\ &= \sum_{k=1}^n \left(|R_i(\omega_k)|^2 - \frac{C + (D - H)\omega_k^2 - 1}{C + (D + H)\omega_k^2 + 1} \right)^2 \end{aligned} \quad (4.61)$$

For the minimum sum of the square error, we can set the derivatives of $E_i(C, D)$ with respect to C and D to zero, respectively. Therefore,

$$\begin{aligned} 0 &= \frac{\partial E_i(C, D)}{\partial C} \\ &= \sum_{k=1}^n \left(\frac{4}{[C + (D - H)\omega_k^2 + 1]^2} \right. \\ &\quad \left. \left(\frac{C + (D - H)\omega_k^2 - 1}{C + (D + H)\omega_k^2 + 1} - |R_i(\omega_k)|^2 \right) \right) \end{aligned} \quad (4.62)$$

and

$$\begin{aligned}
0 &= \frac{\partial E_i(C, D)}{\partial D} \\
&= \sum_{k=1}^n \left(\frac{4\omega_k^2}{[C + (D - H)\omega_k^2 + 1]^2} \right) \cdot \\
&\quad \left(\frac{C + (D - H)\omega_k^2 - 1}{C + (D - H)\omega_k^2 + 1} - |R_i(\omega_k)|^2 \right)
\end{aligned} \tag{4.63}$$

Since the model in Eq. 4.60 is nonlinear, the solution of C , D , and H cannot be obtained directly. However, we can solve for C , D , and H by Newton's methods [74].

From Eqs. 4.62 and 4.63, two functions, $f_1(C, D)$ and $f_2(C, D)$ can be defined as:

$$\begin{aligned}
f_1(C, D) &= \frac{\partial E_i(C, D)}{\partial C} \\
&= \sum_{k=1}^n \left(\frac{4}{[C + (D - H)\omega_k^2 + 1]^2} \right) \cdot \\
&\quad \left(\frac{C + (D - H)\omega_k^2 - 1}{C + (D - H)\omega_k^2 + 1} - |R_i(\omega_k)|^2 \right)
\end{aligned} \tag{4.64}$$

and

$$\begin{aligned}
f_2(C, D) &= \frac{\partial E_i(C, D)}{\partial D} \\
&= \sum_{k=1}^n \left(\frac{4\omega_k^2}{[C + (D - H)\omega_k^2 + 1]^2} \right) \cdot \\
&\quad \left(\frac{C + (D - H)\omega_k^2 - 1}{C + (D - H)\omega_k^2 + 1} - |R_i(\omega_k)|^2 \right)
\end{aligned} \tag{4.65}$$

The solutions of C and D can be obtained when both $f_1(C, D)$ and $f_2(C, D)$ are equal to zero. This result can be found by iteration. Initial values C_0 and D_0 are given and new values C_{k+1} and D_{k+1} obtained from C_k and D_k as follows:

$$\begin{bmatrix} C_{k+1} \\ D_{k+1} \end{bmatrix} = \begin{bmatrix} C_k \\ D_k \end{bmatrix} - J(C_k, D_k)^{-1} \begin{bmatrix} f_1(C_k, D_k) \\ f_2(C_k, D_k) \end{bmatrix} \quad (4.66)$$

where $J(C_k, D_k)$ is the Jacobian matrix of (C_k, D_k) , and defined as:

$$J(C_k, D_k) = \begin{bmatrix} \frac{\partial f_1(C, D)}{\partial C} & \frac{\partial f_1(C, D)}{\partial D} \\ \frac{\partial f_2(C, D)}{\partial C} & \frac{\partial f_2(C, D)}{\partial D} \end{bmatrix}_{C=C_k, D=D_k} \quad (4.67)$$

The algorithm to find solutions of C and D can be described as follows:

Step 1: set $k = 0$, assuming C_0 and D_0 are given initial values;

$$\text{Step 2: } H = \frac{a_i \sqrt{2D_k(C_k \pm \sqrt{C_k^2 - 1})(b_i/\rho_i)^2 - a_i^2}}{b_i(C_k \pm \sqrt{C_k^2 - 1})}$$

$$\text{Step 3: } \begin{bmatrix} C_{k+1} \\ D_{k+1} \end{bmatrix} = \begin{bmatrix} C_k \\ D_k \end{bmatrix} - J(C_k, D_k)^{-1} \begin{bmatrix} f_1(C_k, D_k) \\ f_2(C_k, D_k) \end{bmatrix}$$

Step 4: if $|f_1(C_{k+1}, D_{k+1})| + |f_2(C_{k+1}, D_{k+1})| > \xi$ and $k < k_{max}$

then $k = k + 1$; go to Step 2;

else STOP.

In the above algorithm, the iteration will continue until the value of $|f_1(C_{k+1}, D_{k+1})| + |f_2(C_{k+1}, D_{k+1})|$ is less than a small quantity ξ , at which point a good approximate solution is obtained. To avoid the solutions from local minimum squared error, some other initial values should be tried to make sure the solutions have a global minimum. If the iterative number, k , is equal to the maximum allowable number of iterations, k_{max} , and the value of $|f_1(C_{k+1}, D_{k+1})| + |f_2(C_{k+1}, D_{k+1})|$

is greater than ξ , the iteration may diverge so that a good approximation is not obtained. If an unacceptable solution occurs, other initial values should be tried.

When the solutions of C and D with minimum squared error, $\hat{C}$ and $\hat{D}$, have been found, the velocity-density product and the attenuation-density quotient in the $(i + 1)$ th medium can be expressed in terms of the quantities of the i th layer by:

$$\begin{aligned} v_{i+1}\rho_{i+1} &= \frac{\rho_{i+1}}{b_{i+1}} \\ &= \frac{v_i\rho_i}{[\hat{C} \pm \sqrt{\hat{C}^2 - 1}]} \end{aligned} \quad (4.68)$$

$$\begin{aligned} \frac{\alpha_{i+1}}{\rho_{i+1}} &= \frac{a_{i+1}\omega^2}{\rho_{i+1}} \\ &= \left(\frac{2[\hat{C} \pm \sqrt{\hat{C}^2 - 1}]\hat{D}}{(v_i\rho_i)^2} - \left(\frac{a_i}{\rho_i}\right)^2 \right)^{1/2} \omega^2 \end{aligned} \quad (4.69)$$

where $\hat{C}$ and $\hat{D}$ are the solutions with minimum sum of the square error for C and D . The $\pm$ sign in Eqs. 4.68 and 4.69 is determined by the same rule as for Eqs. 4.53 and 4.54.

It can be seen that the velocity-density product and attenuation-density quotient of the second medium can be found if the parameters of the first medium are known. To generalize the approach, it is possible to evaluate material properties of an n -layered structure by repeating the procedure successively.

4.3 Derivations under the Assumption that the Attenuation Coefficient Is Proportional to the l -th Power of Frequency

For some media, the attenuation constants are not proportional to frequency squared. In this section, a more general case will be discussed. We assume the attenuation constant α is proportional to the l -th power of frequency, and β is still proportional to frequency.

The following equations can be obtained from Eqs. 4.19 and 4.27:

$$\alpha \approx a\omega^l \quad (4.70)$$

$$\beta \approx b\omega \quad (4.71)$$

From 4.25, we have

$$|R_i(\omega)|^2 = \frac{[(b_i/\rho_i) - (b_{i+1}/\rho_{i+1})]^2 + [(a_i\omega^{l_i-1}/\rho_i) - (a_{i+1}\omega^{l_{i+1}-1}/\rho_{i+1})]^2}{[(b_i/\rho_i) + (b_{i+1}/\rho_{i+1})]^2 + [(a_i\omega^{l_i-1}/\rho_i) + (a_{i+1}\omega^{l_{i+1}-1}/\rho_{i+1})]^2} \quad (4.72)$$

$$\frac{1 + |R_i(\omega)|^2}{1 - |R_i(\omega)|^2} = \frac{(b_i/\rho_i)^2 + (b_{i+1}/\rho_{i+1})^2 + (a_i\omega^{l_i-1}/\rho_i)^2 + (a_{i+1}\omega^{l_{i+1}-1}/\rho_{i+1})^2}{2[(b_i/\rho_i)(b_{i+1}/\rho_{i+1}) + (a_i\omega^{l_i-1}/\rho_i)(a_{i+1}\omega^{l_{i+1}-1}/\rho_{i+1})]} \quad (4.73)$$

If we assume the attenuation of the i -th layer is zero, that is, $\alpha_i = 0$, and $a_i = 0$,

then Eq. 4.73 can be simplified to:

$$\frac{1 + |R_i(\omega)|^2}{1 - |R_i(\omega)|^2} = \frac{(b_i/\rho_i)^2 + (b_{i+1}/\rho_{i+1})^2 + (a_{i+1}\omega^{l_{i+1}-1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.74)$$

or,

$$\begin{aligned} F(\omega) &\equiv \frac{1 + |R_i(\omega)|^2}{1 - |R_i(\omega)|^2} \\ &= A + B\omega^m \end{aligned} \quad (4.75)$$

where

$$A \equiv \frac{(b_i/\rho_i)^2 + (b_{i+1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.76)$$

$$B \equiv \frac{(a_{i+1}/\rho_{i+1})^2}{2[(b_i b_{i+1})/(\rho_i \rho_{i+1})]} \quad (4.77)$$

$$m \equiv 2(l_{i+1} - 1) \quad (4.78)$$

The total sum of square error (*SSE*) of *n* data points will be

$$(SSE) \equiv \sum_{i=1}^n [F(\omega_i) - (A + B\omega_i^m)]^2 \quad (4.79)$$

The minimum values of (*SSE*) will occur when $\frac{\partial(SSE)}{\partial A} = 0$, $\frac{\partial(SSE)}{\partial B} = 0$, and $\frac{\partial(SSE)}{\partial m} = 0$

$$\begin{aligned} 0 &= \frac{\partial(SSE)}{\partial A} \\ &= \sum_{i=1}^n (-2)[F(\omega_i) - (A + B\omega_i^m)] \end{aligned} \quad (4.80)$$

$$\begin{aligned}
0 &= \frac{\partial(SSE)}{\partial B} \\
&= \sum_{i=1}^n (-2\omega_i^m) [F(\omega_i) - (A + B\omega_i^m)]
\end{aligned} \tag{4.81}$$

$$\begin{aligned}
0 &= \frac{\partial(SSE)}{\partial m} \\
&= \sum_{i=1}^n (-2B(\ln \omega_i) \omega_i^m) [F(\omega_i) - (A + B\omega_i^m)]
\end{aligned} \tag{4.82}$$

Eqs. 4.80, 4.81, and 4.82 can be written as

$$\sum_{i=1}^n (A + B\omega_i^m) = \sum_{i=1}^n F(\omega_i) \tag{4.83}$$

$$\sum_{i=1}^n (A + B\omega_i^m) \omega_i^m = \sum_{i=1}^n F(\omega_i) \omega_i^m \tag{4.84}$$

$$\sum_{i=1}^n (A + B\omega_i^m) \omega_i^m \ln \omega_i = \sum_{i=1}^n F(\omega_i) \omega_i^m \ln \omega_i \tag{4.85}$$

This provides three unknowns (A, B, m) and three equations (Eqs. 4.83, 4.84, and 4.85). This nonlinear regression problem can be solved for A, B , and m by iteration. First, the two linear parameters A and B can be expressed in terms of $F(\omega_i), \omega_i$, and m . From Eq. 4.83,

$$A = \frac{\sum_{i=1}^n (F(\omega_i) - B\omega_i^m)}{n} \tag{4.86}$$

Inserting Eq. 4.86 into Eq. 4.84, allows B to be expressed as

$$B = \frac{\sum_{i=1}^n \omega_i^m [F(\omega_i) - \sum_{j=1}^n (F(\omega_j)/n)]}{\sum_{i=1}^n \omega_i^m [\omega_i^m - \sum_{j=1}^n (\omega_j^m/n)]} \tag{4.87}$$

From Eq. 4.85 we can define a new function $f(m)$,

$$f(m) \equiv \sum_{i=1}^n [F(\omega_i) - (A + B\omega_i^m)] \omega_i^m \ln \omega_i \quad (4.88)$$

with the solution determined by $f(m) = 0$. Therefore, the problem reduces to finding zeroes of $f(m)$ when A and B are known. First, an initial value of m_0 is assumed, a new value m_1 is found by Newton's method to make $f(m_1)$ approach zero. This method is continued according to the recursion relationship:

$$m_{k+1} = m_k - \left(f(m_k) / \left[\frac{df(m)}{dm} \right]_{m=m_k} \right) \quad (4.89)$$

where $\frac{df(m)}{dm}$ can be obtained from Eq. 4.88 as follows:

$$\frac{df(m)}{dm} = \sum_{i=1}^n [F(\omega_i) - (A + 2B\omega_i^m)] \omega_i^m [\ln \omega_i]^2 \quad (4.90)$$

Therefore, the values of A_k and B_k can be obtained from Eqs. 4.86-4.90.

The algorithm for the nonlinear regression can be described as follows:

Step 1: set $k = 0$, assume an initial value for m_0 ,

$$\text{Step 2: } B = \frac{\sum_{i=1}^n \omega_i^{m_k} [F(\omega_i) - \sum_{j=1}^n (F(\omega_j)/n)]}{\sum_{i=1}^n \omega_i^{m_k} [\omega_i^m - \sum_{j=1}^n (\omega_j^m/n)]}$$

$$A = \frac{\sum_{i=1}^n (F(\omega_i) - B\omega_i^{m_k})}{n}$$

$$\text{Step 3: } m_{k+1} = m_k - \left(f(m_k) / \left[\frac{df(m)}{dm} \right]_{m=m_k} \right)$$

Step 4: if $|f(m_{k+1})| > \xi$ and $k < k_{max}$

then $k = k + 1$; go to Step 2;

else STOP.

Following the discussion in section 4.2, the iteration will continue until the absolute value of $f(m_k)$ is less than a small quantity ξ , at which point a good approximate solution is obtained. To avoid solutions which are from local minimum squared errors, other initial values should be tried to make sure the solutions with global minimum are found. If the iterative number, k , is equal to the maximum allowable number of iterations, k_{max} , and the absolute value of $f(m_k)$ is greater than ξ , the iteration does not converge so that a good approximation is not obtained. If an unacceptable solution results, other initial values should be tried.

CHAPTER 5

EXPERIMENTAL PROCEDURE AND RESULTS

In order to analyze the characteristics of an unknown medium by acoustic waves, we can use either the transmission method or the reflection method. For the transmission method, at least two transducers, one for transmitting and the other for receiving, must be used. In addition, the two transducers must be aligned on the same axis to maximize the detected power. On the other hand, the reflection method only requires one transducer. This transducer acts as both the transmitter and the receiver. Since the reflection mode measurements are easier to implement, this configuration is chosen for experimental confirmation.

Figure 5.1 shows a computer simulation for reflected signals, which are Gaussian pulses modulated with 2.25 MHz, from the interface of water and three different materials. The three materials are aluminum, plexiglass, and wood. There are several facts we can find from the simulation results; first, the amplitude of the reflected signal depends on the impedance of materials. The phase of each signal is also related to the impedance difference at the reflected interface. For example, the phases of reflected signals from aluminum and plexiglass are the same as for the incident signals. This is because the impedances of aluminum and plexiglass are greater than the impedance

of water. On the other hand, the impedance of wood is less than the impedance of water, therefore the phase of the reflected signal from the interface of water and wood will be different than the phase of the incident signal. Secondly, the reflected signal may not be as symmetrical as the incident signal because of the dispersive properties of the material. However, the shape of the reflected signal contains the characteristics of the materials and provides information for material feature identification.

5.1 Experimental Procedure

An essential part of the measurement procedure is to record the detailed time and frequency characteristics of the incident broadband signal, since the reflected wave contains all frequency components modified by their individual frequency dependent reflection coefficients. Experimentally, the incident signal can be calibrated by using a water/air interface which gives almost complete reflection. The effects of diffraction are reduced by using the water/air interface as a reference.

Replacing the water/air interface with a water/object interface will allow investigation of the detailed reflection spectral characteristics for feature identification. The block diagram of the experimental arrangement for reflected mode are shown in Figure 5.2

The incident pulse is obtained from a 2.25 MHz transducer (Panametrics V306) and a Panametrics 5050 pulser, which provides an approximately Gaussian shaped pulse of $.5 \mu\text{sec}$ pulse width. The reflected signal is sampled by an A/D converter with a sampling rate of 40 MHz and 8-bit resolution. The test materials used are aluminum, composite material, plexiglass, and de-gassed wood. For each test material, 256 signals are transmitted and the average of those reflected signals is recorded with

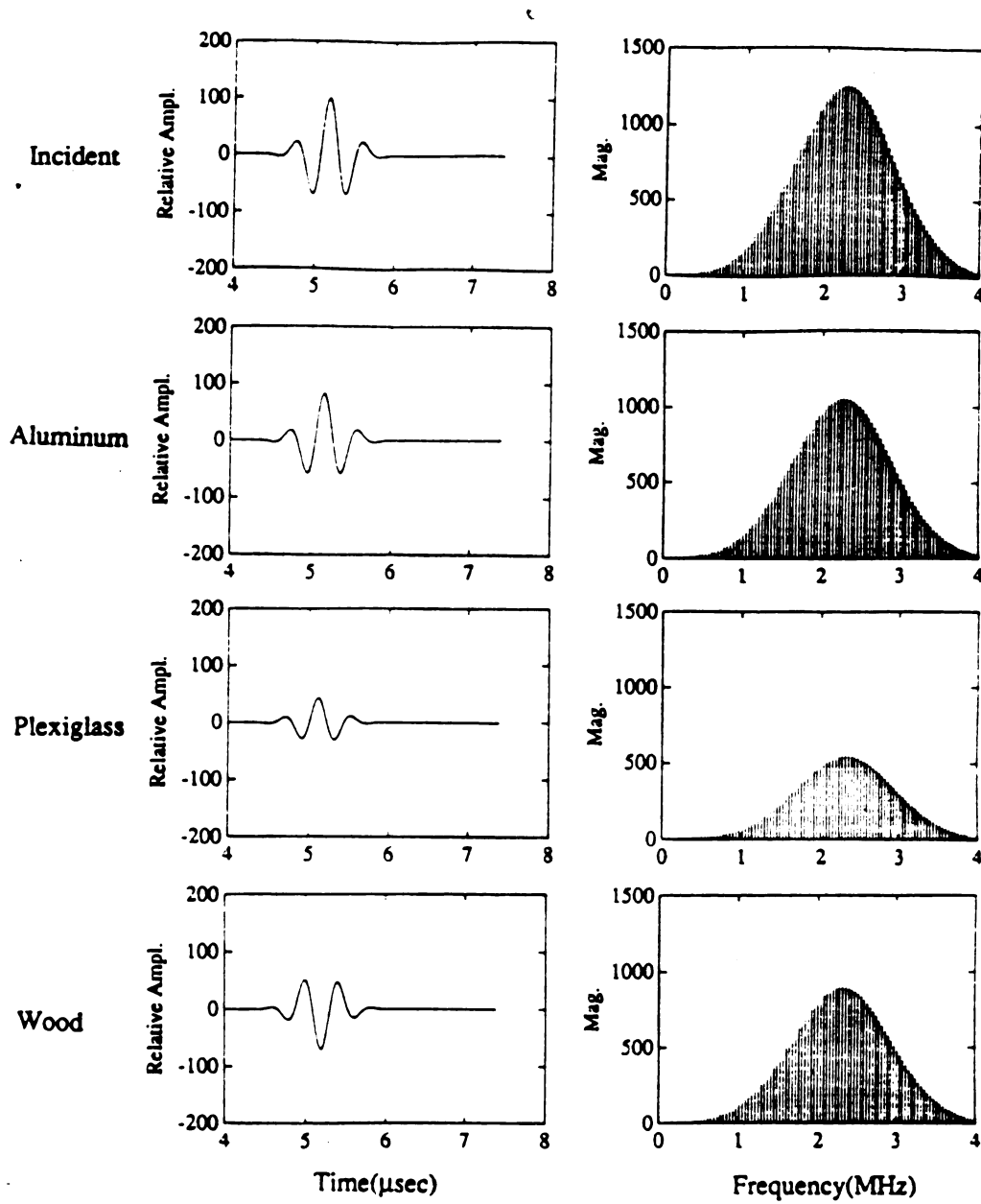


Figure 5.1. Computer simulation of the reflected Gaussian pulse from four different materials

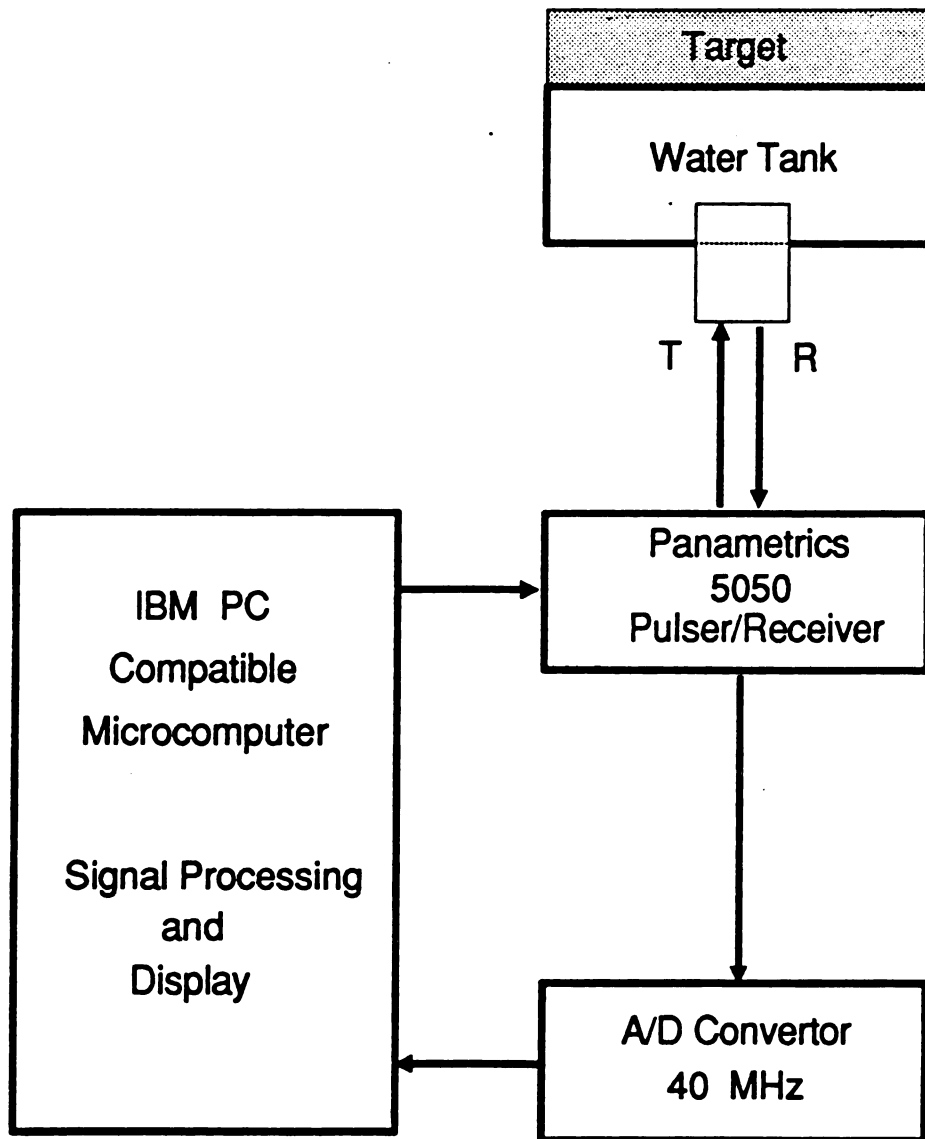


Figure 5.2. The block diagram of the experimental arrangement for reflected mode

1024 sampling points. From the average of 256 signals, the signal-to-noise ratio was improved by a factor of $\sqrt{256}$, or 16. Figure 5.3 shows the procedure of a computer program which will control the sampling unit and take the average of the reflected signals. This program is written in C language. The starting sampling points, the number of sampling points, the trigger level, and the number of signals to be averaged, are set at program initiation. The averaged signal is displayed on a monitor screen and the value of each sampling point is stored in a computer memory data file.

Figure 5.4 shows the procedure for data processing. Initially, the incident signal and the reflected signal are recorded in the time domain. After acquiring these signals, the spectra of the incident signal and of the reflected signal can be obtained by a 1024-point Fast Fourier Transform done in software. From the spectra of the incident and reflected signals, the reflection coefficient, $R(\omega)$, of each frequency in the 3 dB bandwidth can be obtained. Using a least-squares regression procedure, the velocity-density product and attenuation-density ratio for each material can be determined.

5.2 Experimental Results

Using the experimental setup mentioned in the previous section, the incident signal and reflected signal from the test materials can be measured. The signals in the time domain and their frequency spectra are shown in Figures 5.5 and 5.6. The measured incident signal in the time domain is close to the Gaussian pulse which is used in computer simulation. As we predicted, the phase of the reflected signals from the interfaces of water and aluminum, composite material, and plexiglass are the same as for the incident signal, while the phase of the reflected signal from the interface of water and wood is different. The spectrum of the incident signal although not a

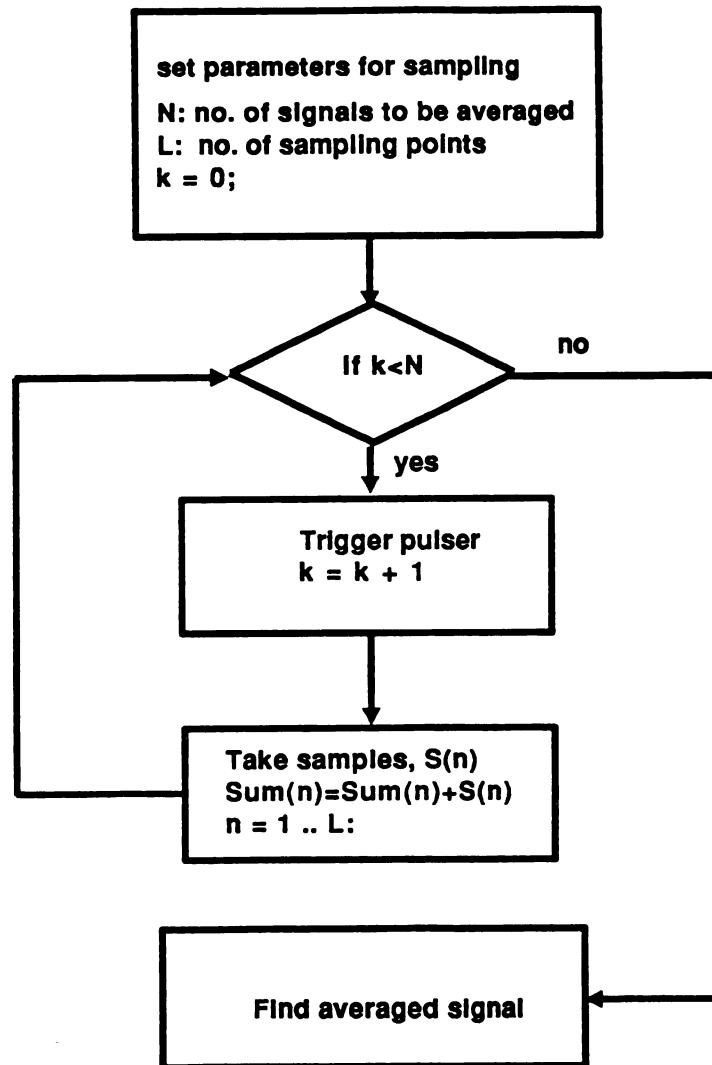


Figure 5.3. The procedure of the sampling program

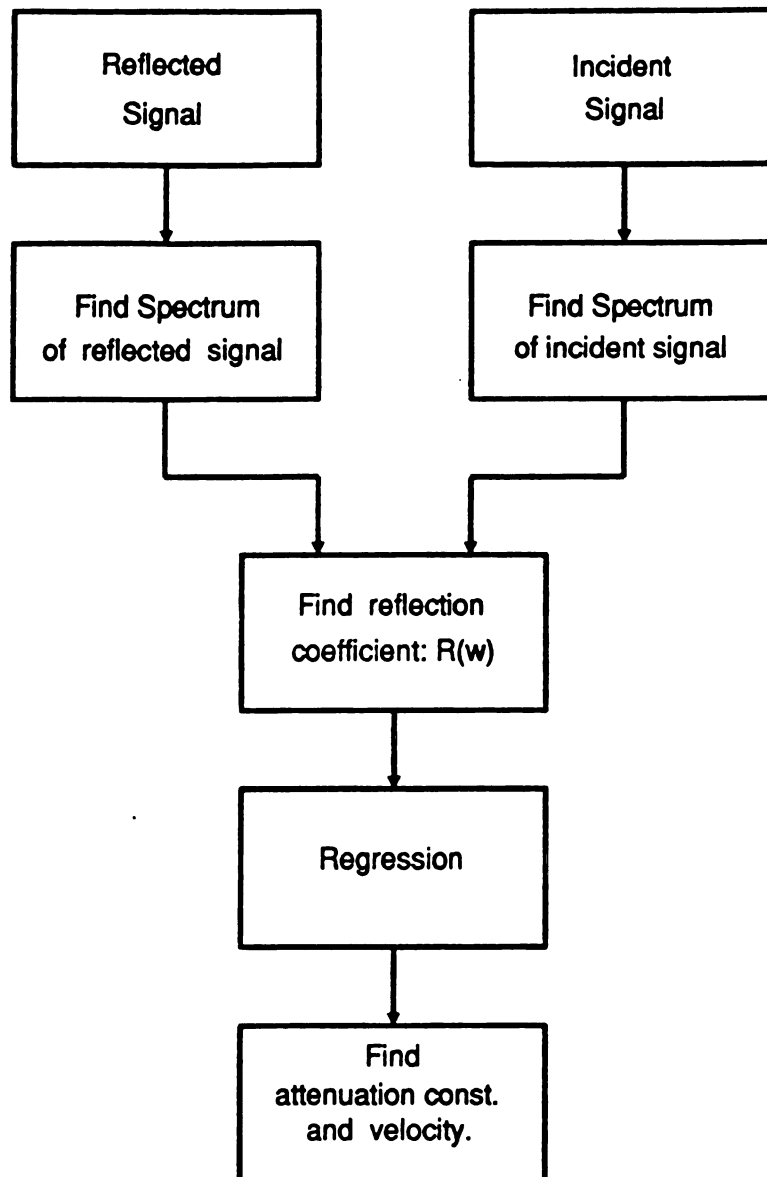


Figure 5.4. The procedure for data processing

perfect Gaussian is sufficiently close that the approximation of a Gaussian shaped spectrum is quite acceptable.

From the spectra of the reflected signal and incident signal, the reflection coefficients in the 3-dB bandwidth can be obtained. Figure 5.7 shows the reflection coefficients, $R(\omega)$, for the interfaces between water and four test materials. From Figure 5.7 we can see that all the reflection coefficients of the test materials have the tendency to increase as frequency increases. This matches the theoretical derivation shown in Eq. 4.32.

From the reflection coefficients in the 3-dB bandwidth, the values of A and B in Eq. 4.35 can be obtained for each material. The results of regression, with 90% prediction confidence intervals, for the above materials can be found from Eqs. 4.51 and 4.52 as shown in Figure 5.8. In the experiments, the 3-dB bandwidth is about 1 MHz and the variation of the reflection coefficients of the test materials is about 0.03 from 1.7 MHz to 2.6 MHz. From another point of view, the reflection coefficient of aluminum is the largest and the reflection coefficient of plexiglass is the smallest for the test materials studied.

Based on the results of Figure 5.8, the velocity and attenuation of test materials can be obtained. The standard deviation of the velocity can also be obtained from Eqs. 4.51 and 4.52. Table 5.1 summarizes the experimental results for a number of different materials using this approach. The values of $\mathcal{R}^2$ statistics of regression are also calculated according to the constraints of the regression. By comparing the experimental results with the published values, it is observed that the velocities of aluminum and plexiglass are very close to the published values in [71, 4]. The variation between the experimental mean values and the published values is less than 3%.

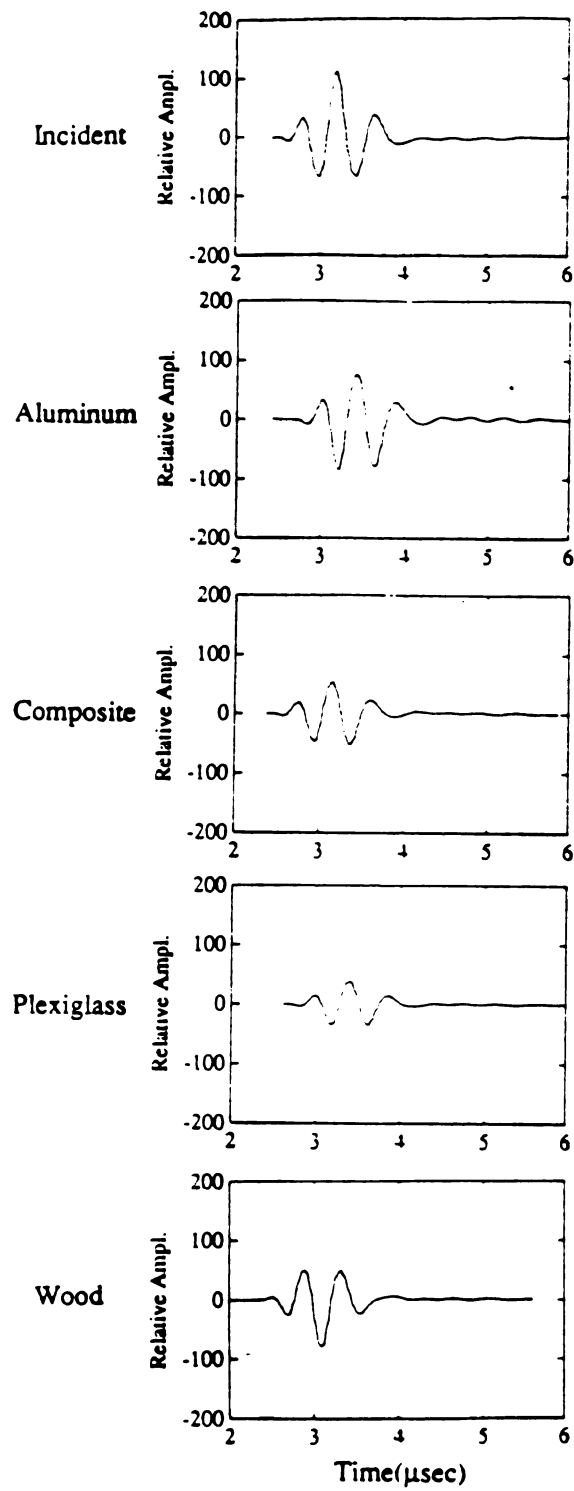


Figure 5.5. The incident signal and reflected signals from test materials

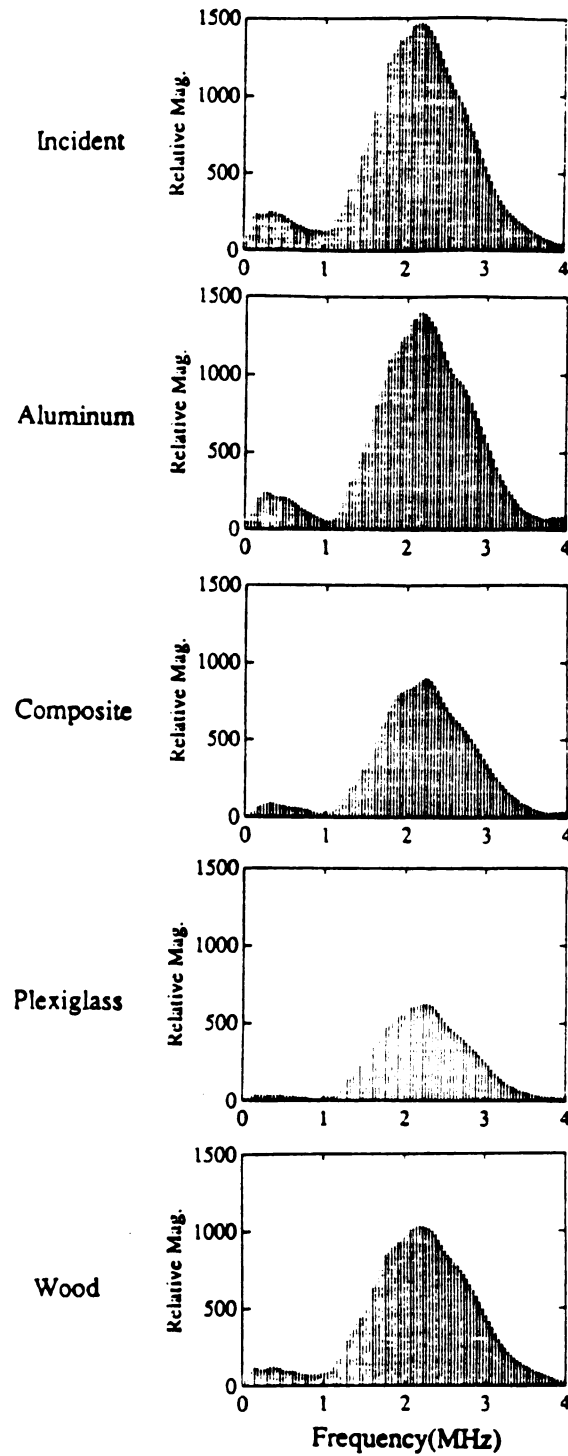


Figure 5.6. The spectra of the incident signal and reflected signals from test materials

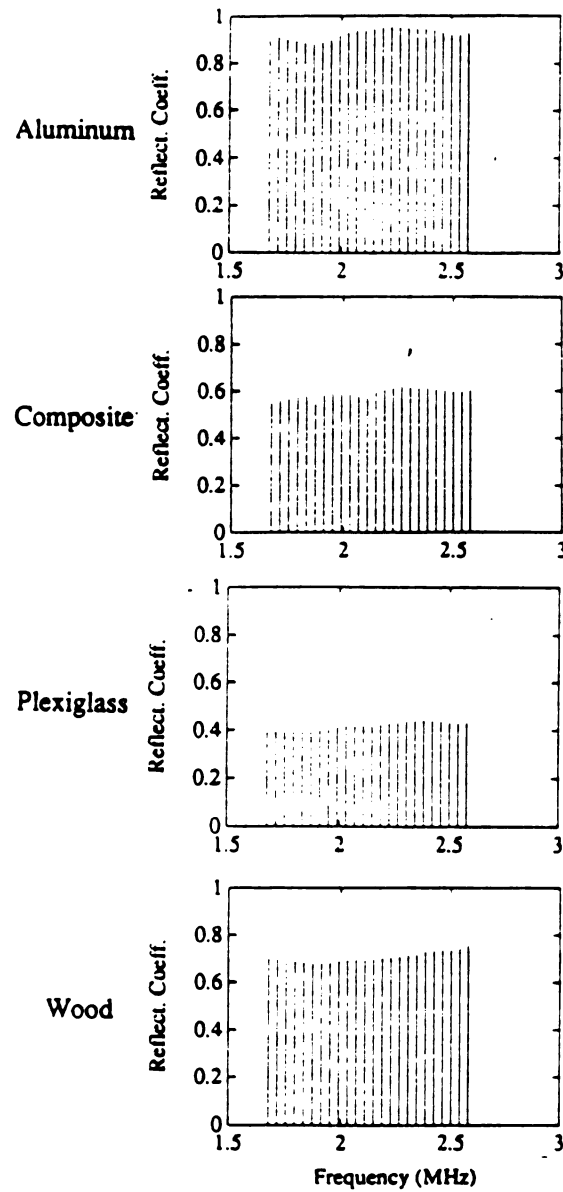


Figure 5.7. The reflection coefficients of test materials within their 3-dB bandwidth

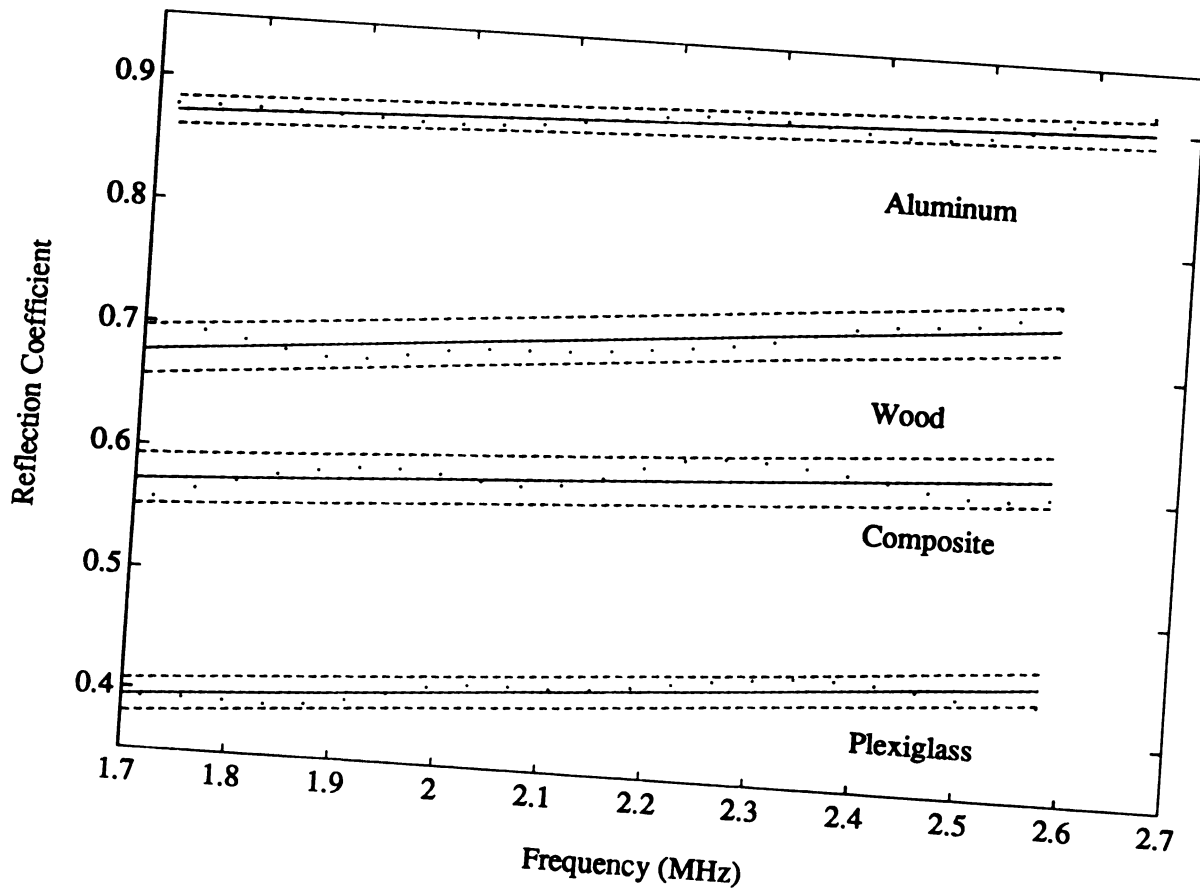


Figure 5.8. The results of regression with 90% prediction confidence interval

Material	Aluminum	Composite	Plexiglass	Wood
Density (kg/m^3)	2695	1609	1182	525
Velocity-density product ($kg \cdot sec^{-1} \cdot m^{-2}$)	1.695×10^7	4.884×10^6	3.075×10^6	3.687×10^5
Attenuation-density ratio (m^2/kg)	6.575	4.238	3.866	25.92
Velocity, <i>Mean</i> $\pm$ <i>S.D.</i> (m/sec)	6287 ± 117	3036 ± 44	2602 ± 29	702 ± 13
Velocity (<i>published</i>) (m/sec)	6350 [71]	—	2670 [4]	—
$\mathcal{R}^2$ Statistics of regression	0.63	0.60	0.82	0.78

Table 5.1. Experimental results of measuring velocity and attenuation by broadband signals

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

5.3 Two Dimensional Imaging by Using Velocity-Density Product

To improve on the measurement results of a one-dimensional experimental setup, a scanning system was built to obtain two-dimensional images. Figure 5.9 shows the block diagram for this scanning system. The system is controlled by an IBM PC/XT compatible computer. Two stepper motors are used to locate the transducer in the proper position. A Panametrics 5050 pulser/receiver is used to excite the transducer and receive reflected signals. Again, the received signal is sampled at 40 MHz and the samples are stored in the computer. By using the technique of broadband signals the velocity-density product of materials can be obtained and the two dimensional image can be constructed by scanning. A computer program written in C language is developed to control the scanning motor, get sampling data, and display color images. The program list of the main program is in Appendix A. This program can get data from the scanning system which is described in Figure 5.9, get data from a previous recorded data file, or display a constructed image file. This program will automatically send out control signals to operate the scanning system according to the preset values. The data obtained from each (X, Y) location in the scanning system can be stored in a file for further analysis. Once the desired feature, such as velocity or attenuation, has been obtained, a two dimensional color image can be displayed on a monitor screen. The colors of images can be selected from 16 preset colors to get the best display.

Figure 5.10 shows a reference sample to demonstrate the performance of the two dimensional scanning system. The sample is made from two concentric cylinders of Teflon and aluminum, respectively. This sample is placed inside a water tank and the ultrasonic transducer scans the top of the sample. Therefore, this represents two

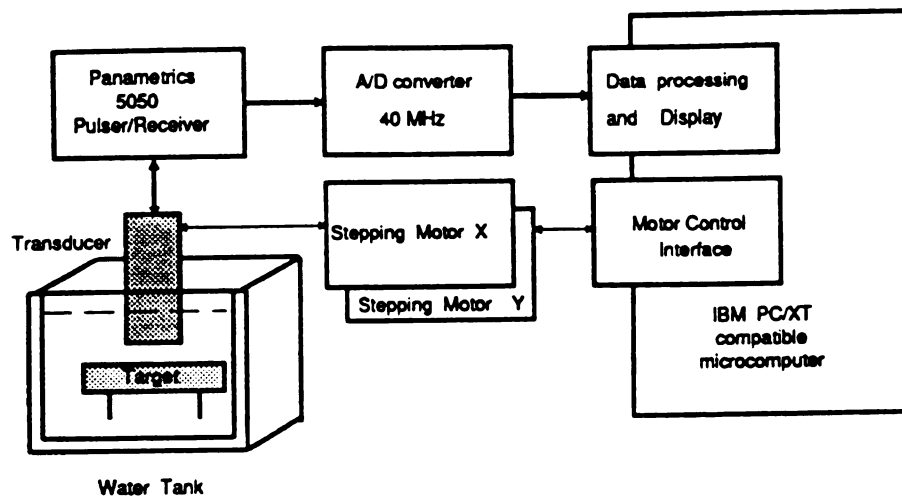


Figure 5.9. The block diagram of the scanning system

cont

Inde

bers

sign

that

the

the

the

have

to d

target

layer

ima

The

each

the

can

is f

contiguous 2-layered sample. The first layer is water and the second layer is either Teflon or aluminum. Using the techniques outlined above, an image of the velocity-density product, as shown in Figure 5.11, is obtained. The blue area is the reflected signal from Teflon and the red area is the reflected signal from aluminum. It is clear that two different materials are present in the image.

A 3-layered model, as shown in Figure 5.12, consisting of a layer of plexiglass placed over the first sample was also investigated. The first layer is water, the second layer is plexiglass, and the third layer is Teflon or aluminum. This model is used to demonstrate that the technique can be applied to retrieve image of deep-lying targets. After evaluating the properties of the first layer, the properties of the second layer can be obtained by using Eqs. 4.53, 4.54, and 4.68. Figure 5.13 shows the image obtained from the velocity-density product for material in the second layer. This image is not as clear as the image obtained from the 2-layer model because the calculation error will accumulate from layer to layer. Although the error is larger than the error in the previous model, the areas reflected from Teflon and aluminum still can be distinguished. This result demonstrates that multilayer material identification is feasible using the techniques described in this research.

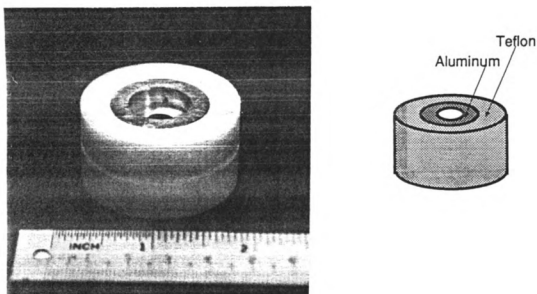


Figure 5.10. A 2-layered 2-dimensional sample model

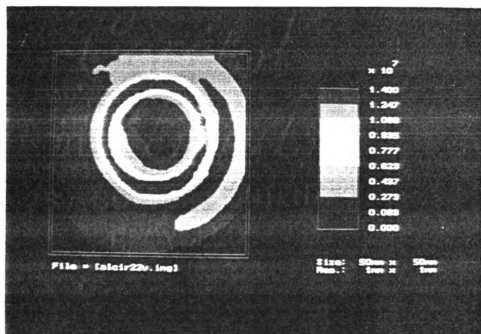


Figure 5.11. Imaging of the 2-layered 2-dimensional sample model using velocity-density product data

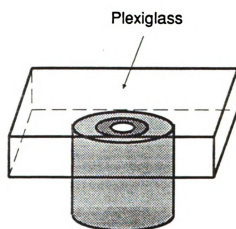
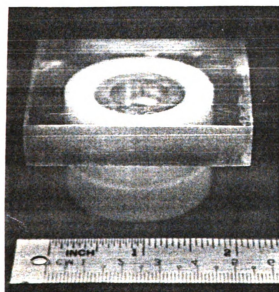


Figure 5.12. A 3-layered 2-dimensional sample model

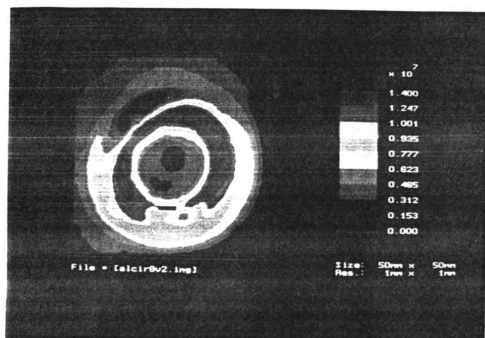


Figure 5.13. Imaging of the 3-layered 2-dimensional sample model using velocity-density product data

CHAPTER 6

ULTRASONIC IMAGES AND TISSUE CHARACTERIZATION BY HIERARCHICAL CLUSTERING TECHNIQUES

In the previous chapter the information of velocity and attenuation of test materials has been determined by broadband signals. In addition to the one-dimensional processing, two-dimensional images using velocity-density product have been constructed. In order to investigate the properties of materials (or tissue), classification techniques are used for automation. The objective of this chapter is to classify images constructed by ultrasonic broadband signals using hierarchical clustering methods [75, 76]. Section 6.1 introduces the hierarchical clustering techniques. In section 6.2 the image from the velocity-density product obtained from a 2-layered model is used for classification. The clustering results for this known model provides the confidence required to classify images using hierarchical clustering methods. Finally, the clustering technique is applied to a section of human brain with a hemorrhaged tumor. in section 6.3.

6.1 Hierarchical Clustering

A hierarchical clustering is a sequence of partitions in which each partition is nested into the next partition in the sequence [76]. This method provides techniques to show the relationship among test patterns. In this research, the test pattern will be obtained from the value of each pixel in an ultrasonic image. By this technique the patterns with similar properties will be classified as in the same group. Therefore, the patterns with different properties will be distinguished and material identification or tissue characterization can be done. Before using a hierarchical clustering method to classify images some matrices and preprocessing should be performed.

Pattern Matrix

The first step in clustering data is to construct the pattern matrix $\mathcal{P}$. At each (X, Y) location of the test sample several features, such as total energy, peak frequency, or velocity-density product, can be measured. Once the information of each feature is obtained a pattern matrix can be construct for further analysis. A pattern matrix is a matrix which contains the information of each feature in a column and all information obtained from a certain location in one row.

Normalization

Since the scale and units of each feature are different, all features must be normalized before processing. The normalization is based on the mean and variance of each feature. If the number of patterns in the analysis is n , the j -th feature value for the i -th pattern denoted by x_{ij}^* , the data set can be normalized by:

$$x_{ij} = \frac{x_{ij}^* - m_j}{s_j} \quad (6.1)$$

where

and

2, 1

is the

as 1

Eig

for

the

as

w

1

where

$$m_j = \frac{\sum_{i=1}^n x_{ij}^*}{n} \quad (6.2)$$

and

$$s_j^2 = \frac{\sum_{i=1}^n (x_{ij}^* - m_j)^2}{n} \quad (6.3)$$

x_{ij} is the normalized value of x_{ij}^* , m_j is the sample mean of the j -th feature, and s_j^2 is the sample variance of the j -th feature. The normalized pattern matrix is denoted as $\mathcal{P}_N = [x_{ij}]$.

Eigenvector Projection

Since the features may be correlated, the normalized pattern matrix should be further transformed to uncorrelated features by eigenvector projection. This projection is performed by the eigenvectors of the covariance matrix $\mathcal{C}$, which is defined as:

$$\mathcal{C} = \frac{\mathcal{P}_N^T \mathcal{P}_N}{n} \quad (6.4)$$

where $\mathcal{P}_N^T$ is the transpose matrix of $\mathcal{P}_N$.

Finding the eigenvectors of the covariance matrix $\mathcal{C}$, the pattern matrix with uncorrelated features, $\mathcal{P}_U$, can be found.

$$\mathcal{P}_U = \mathcal{P}_N \begin{bmatrix} \mathcal{V}_1^T \\ \mathcal{V}_2^T \\ \cdot \\ \cdot \\ \cdot \\ \mathcal{V}_d^T \end{bmatrix}^T \quad (6.5)$$

where $\mathcal{V}_i$ is the i th eigenvector of the covariance matrix, d is the total number of features.

Proximity Matrix

A proximity matrix should be constructed by using the pattern matrix with uncorrelated features. The proximity matrix contains information on the Euclidean distance of each pair of patterns. The (i, j) -entry in the proximity matrix provides the distance between pattern i and pattern j . This results in a symmetrical proximity matrix. The size of a proximity matrix of an image with n pixels is n by n , which is usually very large. For example, a 50 by 50 image will contain 2601 pixels, and the proximity matrix will be 2601 by 2601.

Hierarchical Clustering Methods

The most commonly referenced hierarchical clustering methods are [76]:

1. Single link method.
2. Complete link method.
3. Ward's method.
4. Unweighted Pair Group Method using Arithmetic averages (UPGMA).

For the single link and the complete link methods, the clustering results depend on the proximities through their rank order only. Ward's method provides a minimum square error and is therefore also called the minimum variance method. UPGMA (Unweighted Pair Group Method using Arithmetic averages) computes the distance between two groups by arithmetic averages. The effectiveness of each of the above algorithms is discussed below:

Assume the proximity matrix is $\mathcal{D} = [d(i, j)]$, and there are n patterns. $L(m)$ is the distance of clusters after the m th merging operation. The algorithms each follow the initial five step sequence:

Step 1: Set level $L(0) = 0$ and the sequence number $m = 0$.

Step 2: Find the cluster with the minimum distance between each pair.

Assume the minimum distance pair is cluster r and cluster s .

The distance between cluster r and cluster s is

$$d[(r), (s)] = \min\{d[(i), (j)]\}$$

Step 3: Increment the sequence number, $m = m + 1$.

Set the m th level $L(m) = d[(r), (s)]$.

Step 4: Calculate the distance between the old cluster k and

the new cluster (r, s) , $d[(k), (r, s)]$, in

the proximity matrix, $\mathcal{D}$.

Step 5: After all objects are in one cluster then STOP,

else go to Step 2.

The distance between the old cluster k and the new cluster (r, s) , $d[(k), (r, s)]$, is calculated according to the different clustering methods as follows:

for the single link method:

$$d[(k), (r, s)] = \min\{d[(k), (r)], d[(k), (s)]\}, \quad (6.6)$$

for the complete link method:

$$d[(k), (r, s)] = \max\{d[(k), (r)], d[(k), (s)]\}, \quad (6.7)$$

for the Ward's method:

$$d[(k), (r, s)] = \frac{1}{(n_r + n_s + n_k)} [(n_r + n_k)d[(k), (r)] + (n_s + n_k)d[(k), (s)] - n_k d[(k), (r)]], \quad (6.8)$$

and for the UPGMA:

$$d[(k), (r, s)] = \frac{n_r d[(k), (r)] + n_s d[(k), (s)]}{(n_r + n_s)}. \quad (6.9)$$

where n_r , n_s , and n_k are the number of patterns in cluster r , s , and k , respectively.

Cluster Validation

After getting the results of clustering, a quantitative and objective method is usually necessary to evaluate the results of clustering. The discussion of cluster validation provides an index of the accuracy of clustering. Usually, the validity of a clustering result can be expressed by the following three indices:

1. External index: a measurement by matching a clustering result to *a priori* information.
2. Internal index: a measurement by using the original data only, no *a priori* information is needed.
3. Relative index: comparison of two or more clustering results to decide which one is more appropriate for the data.

Since there is no *a priori* information on the human brain sample in our experiments, only the internal index and the relative index will be discussed. The most common internal index is called CPCC, or cophenetic correlation coefficient. Be-

fore defining the CPCC, the cophenetic proximity matrix must be constructed. The cophenetic proximity matrix is a matrix which contains the clustering results from a certain method. The (i, j) entry of the cophenetic proximity matrix is the level of object i and object j merged into the same group.

The definition of the CPCC is:

$$\text{CPCC} = \frac{(1/M) \sum_{i=1}^n \sum_{j=1}^n d(i, j) d_c(i, j) - (m_D m_C)}{[(1/M) \sum_{i=1}^n \sum_{j=1}^n d^2(i, j) - m_D^2]^{1/2} [(1/M) \sum_{i=1}^n \sum_{j=1}^n d_c^2(i, j) - m_C^2]^{1/2}} \quad (6.10)$$

where

$$M = \frac{n(n-1)}{2} \quad (6.11)$$

$$m_D = \left(\frac{1}{M}\right) \sum_{i=1}^n \sum_{j=1}^n d(i, j) \quad (6.12)$$

$$m_C = \left(\frac{1}{M}\right) \sum_{i=1}^n \sum_{j=1}^n d_c(i, j) \quad (6.13)$$

$d(i, j)$ is the given proximity between objects i and j , and $d_c(i, j)$ is the cophenetic proximity.

The value of CPCC is between -1 and 1. When CPCC is closer to 1, it means the clustering result fits the original data better. If CPCC is equal to 1, it means the clustering is perfectly matched to the original data, whereas, if CPCC is equal to -1, the clustering result indicates the data is totally mismatched. Once the internal index for each method is obtained, the values can be used as a relative index to compare the clustering results between the different methods.

6.2 Material Identification by Ultrasonic Images

Material identification is an important topic in ultrasound. By using broadband signals, more information can be obtained in one transmitted signal than by conventional detection techniques. In previous chapters, it has been proved that the velocity-density product and attenuation-density ratio can be obtained from broadband signals. By using ultrasonic scanning systems two-dimensional images can be constructed. In this section, the images obtained from a known sample will be used to identify test materials. The sample used in the research is the same as the sample used in section 5.3, which is shown in Figure 5.10. Using the image of velocity-density product as shown in Figure 5.11, material identification can be done by hierarchical clustering methods. In the image of Figure 5.10, only the velocity-density product feature of the material is shown. Because the size of the image is 51 by 51 and there are only two different materials in this model, we can consider this problem as classifying 2601 patterns into two groups. A pattern matrix of 2601 by 1 and a proximity matrix of 2601 by 2601 are constructed before clustering. Since there is only one feature, it is not necessary to do normalizations and eigenvector transformations. By the single link, complete link, Ward's, and UPGMA methods, which have been mentioned in section 6.1, all the patterns will be classified into proper groups.

Once the results of clustering have been obtained, color images can be constructed and show the areas of different materials. The results of the single link, complete link, Ward's, and UPGMA methods are shown in Figures 6.1, 6.2, 6.3, and 6.4, respectively. The results obtained from the complete link, Ward's, and UPGMA methods are fairly similar except for the boundary between the Teflon and the aluminum. All three methods provide clear and good results to show two different materials from the images. The single link method classifies the area reflected from aluminum by two subgroups while the area reflected from Teflon is still in one cluster. Therefore, a

three-cluster image is shown instead of a two-cluster image.

In order to test the validation of clustering by different methods the values of CPCC are calculated according Eq. 6.10. The values of CPCC by different clustering methods for the 2-layered model are shown in Table 6.1 and Figure 6.5.

Method	Value
Single Link	0.50
Complete Link	0.50
Ward's method	0.52
UPGMA	0.81

Table 6.1. The values of CPCC by different clustering methods for the 2-layered model

From Table 6.1 the values of CPCC by the single link, complete link, and Ward's method are very close, which imply that the results of these three methods provide about the same degree of accuracy. According to the values of CPCC, the best clustering result should be the image constructed by UPGMA because its CPCC is the highest. On the other hand, comparing the original model, which is shown in Figure 5.10, and the clustering results, we also find that all the images constructed by the above clustering methods provide good classification.

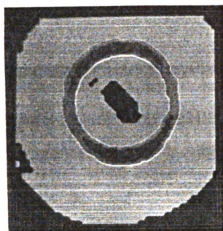


Figure 6.1. Clustering result of a simple 2-layered model by single link clustering method with 3 clusters

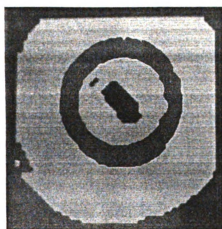


Figure 6.2. Clustering result of a simple 2-layered model by complete link clustering method with 2 clusters

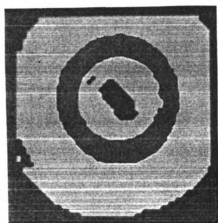


Figure 6.3. Clustering result of a simple 2-layered model by Ward's clustering method with 2 clusters

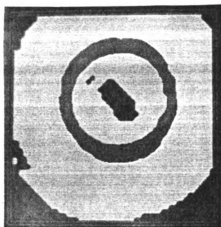


Figure 6.4. Clustering result of a simple 2-layered model by UPGMA clustering method with 2 clusters

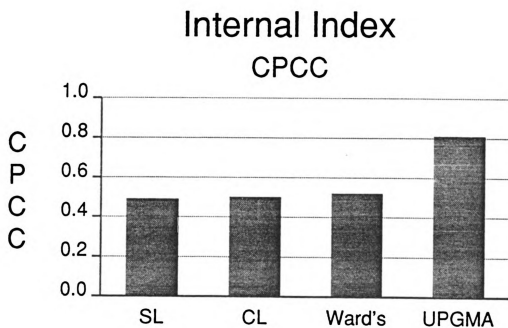


Figure 6.5. Comparison of CPCC by the different methods for the 2-layered model

6.3 Tissue Characterization by Ultrasonic Broad-band Signals

Presently, x-rays are one of the most widely used methods of flaw detection. Because of the potential cumulative risk from body cell ionization use of radiation must be limited and thus ultrasonic diagnosis is preferable. Ultrasonic imaging provides anatomic information based mostly on the differences in impedance at the interface. The echo amplitude of ultrasound is directly proportional to the difference in acoustic impedance across the boundary. Unfortunately, it has been shown that there is no significant variation of acoustic impedance between normal and cancerous tissues [77]. Thus it is extremely difficult to differentiate between cancerous tissue and normal tissue. On the other hand, tumors have quite different attenuation characteristics and propagation velocity.

In order to identify cancerous tissue, additional features must be extracted from the reflected signals, such as velocity, attenuation, peak frequency, bandwidth, etc. Once these features have been extracted, the problem is to process the information to provide tissue classification. Pattern recognition and segmentation are two techniques used for this classification.

6.3.1 Ultrasonic Imaging of Tissues

Since most ultrasonic imaging systems utilize echo returns from boundaries of different acoustic impedance, the image shows only the outline of the boundaries rather than the material properties. However, since the echo return contains not only boundary information but also the acoustic wave propagation dispersion and attenuation which are both frequency dependent, techniques for evaluation of acoustic velocity and attenuation of a single medium have been investigated [10]. Typically, a sequence

of measurements are made at discrete frequencies over the range of interest [16]. The procedure is quite tedious and time consuming, so instead of using discrete frequencies for each echo measurement, a narrow broad bandwidth pulse is used to investigate the properties of tissue. Each frequency component of the transmitting signal spectrum propagates through the material with a different velocity and attenuation depending on the properties of the medium. Consequently, comparing the pulse shape of the echo return to that of the known incident pulse will allow us to identify the target material having different properties such as density, velocity and attenuation.

The sample used for experimental demonstration consists of a section of human brain with a hemorrhaged tumor [75], as shown in Figure 6.6. This sample is fixed in Formalin and packed inside a plastic bag. The hemorrhaged part of the brain is in the upper left corner of the sample.

In order to evaluate the sample the ultrasonic scanning system previously described, and shown in Figure 5.9, is used. A Panametrics 5050 pulser/receiver is used to excite the transducer and receive reflected signals. The received signal is sampled at 40 MHz with 8-bit resolution. The scanning area is 100 mm by 100 mm, and the scanning separation is 2 mm in both the X axis and the Y axis. For each position, 400 sampling points are recorded and the sampling data is stored in the computer for further analysis. Since the original images have some noise and low resolution, some image processing techniques are used to enhance the images. The details of the feature selection and image enhancement will be discussed next.

6.3.2 Feature Selection

In order to obtain maximum information from a sample, five features from the ultrasonic echo return are extracted; they are the total energy, central frequency, frequency at which the peak amplitude occurs, 3-dB bandwidth of the echo spectrum,

and the correlation coefficient between the incident and reflected signals. These features contain not only the information on acoustic impedance variation but also the attenuation and velocity characteristics.

Total energy

The first feature used is the total energy which can be obtained from a time domain signal $s(n)$. We can define the total energy as the summation of the square of each sampling value.

$$\text{Total Energy} = \sum_{i=1}^N |s(i)|^2 \quad (6.14)$$

The total energy of the reflected signal is related to the reflection coefficient, which contains information on the impedance of the medium. Therefore, the total energy will provide us with information on the impedance of the tissue. An image constructed by utilizing total energy is shown in Figure 6.7. From Figure 6.7 we can see the shape of the brain is very close to the sample. The gray matter of the brain is displayed as dark blue in the image.

Central frequency, frequency with peak amplitude, and bandwidth

It has been shown that the central frequency and frequency with peak amplitude of reflected signals will shift according to the attenuation coefficient of the material [22]. The bandwidth of the reflected signal will also change. Figure 6.8 shows these three features in a typical ultrasonic spectrum, obtained from the Fast Fourier Transform. The central frequency is determined by the center of the 3-dB bandwidth. These three features will provide information on the attenuation properties inside the tissue. The images constructed by these three features are shown in Figures 6.9, 6.10, and 6.11. For a symmetrical spectrum the central frequency will be close to the peak



Figure 6.6. A section of human brain with hemorrhaged tumor



Figure 6.7. Ultrasonic imaging constructed using the total energy of the reflected signals

frequency, but for a typical ultrasonic signal, the spectrum is not symmetrical, so that the central frequency and the peak frequency are not necessarily equal. The three features are fairly closely correlated since the central frequency is determined by the center of the 3-dB bandwidth. From the images obtained from the peak frequency and 3-dB bandwidth, the gray matter is not as clear as the image obtained from total energy, but the upper part and upper left corner did show a difference with respect to other areas. This is due to the difference in the attenuation coefficient of tissue close to the tumor versus the attenuation coefficient in other areas. In the image obtained from the central frequency, the area of gray matter is still clear, but other areas are not as well defined.

Correlation coefficient between the incident and reflected signal

The correlation feature shows the difference between the incident signal and the reflected signal in the time domain. From the shape of the reflected signal the properties of the medium, such as elasticity, stiffness constant, velocity, and attenuation can be determined. The image constructed by the correlation feature is shown in Figure 6.12. The image obtained from the correlation coefficient is close to the image obtained from the total energy. The shape and the gray matter is very clear in this image.

6.3.3 Image Enhancement

Noise represents a major problem for generating ultrasonic images. In order to reduce the effect of noise, a modified median filter has been used to reduce the noise in images. This two-dimensional modified median filter has been found to be very powerful in removing noise from two-dimensional signals without blurring the edges [62].

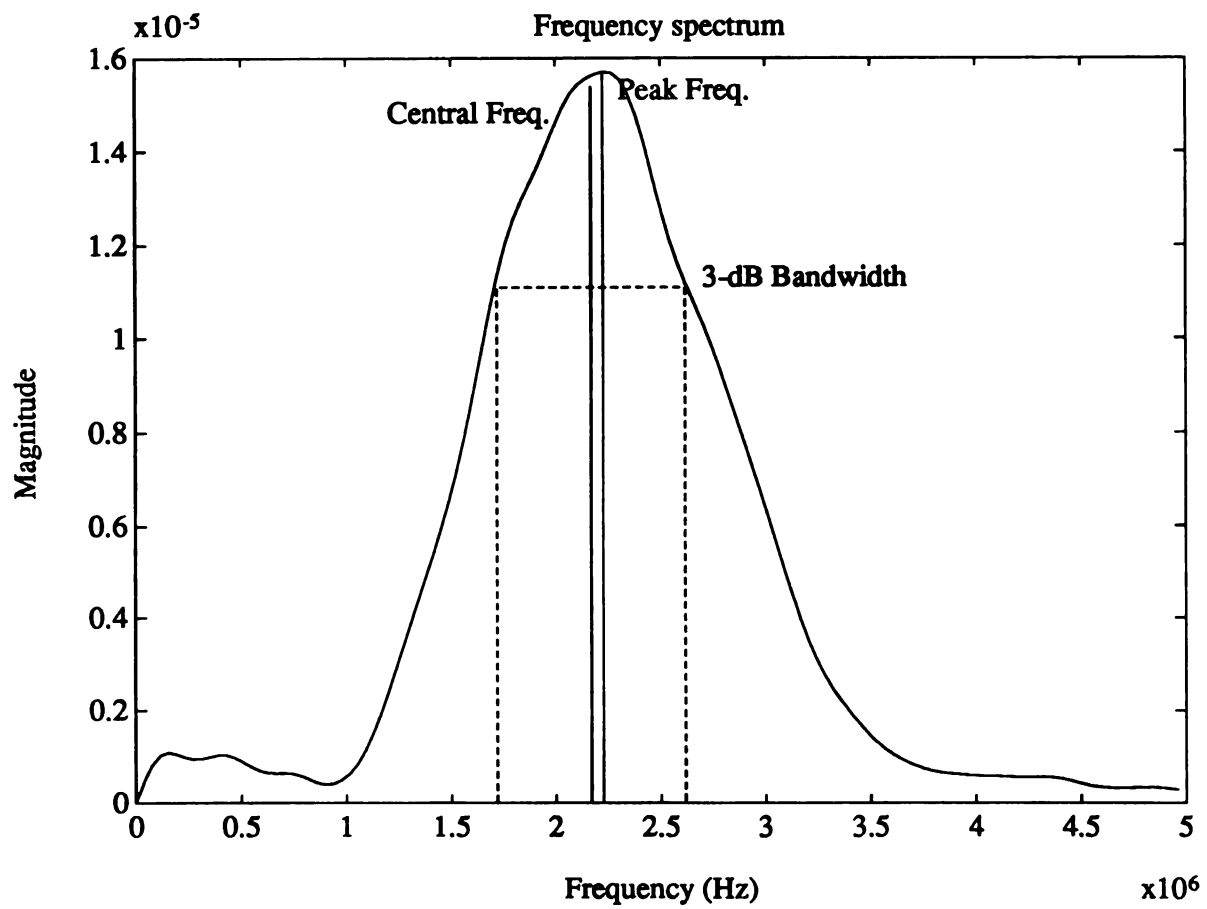


Figure 6.8. Central frequency, peak frequency, and bandwidth of a typical ultrasonic spectrum



Figure 6.9. Ultrasonic image constructed using the peak frequency of the reflected signals



Figure 6.10. Ultrasonic image constructed using the central frequency of the reflected signals



Figure 6.11. Ultrasonic image constructed using the 3-dB bandwidth of the reflected signals

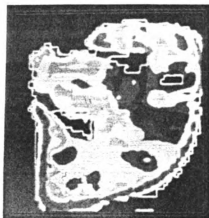


Figure 6.12. Ultrasonic image constructed using the cross correlation of the incident signal and the reflected signals

In computing the median, the set of pixels is restricted to those with a difference in gray level no greater than some threshold.

For a high resolution of the images, a two-dimensional interpolation program has been used to enhance the images. This technique will keep the smooth edges while magnifying images. Assume we have an R by C image, and the intensity of pixel (i, j) is $A_{i,j}$. If the size of the magnified image is (RK_r) by (CK_c) , the intensity of the new image, $B_{i \cdot K_r + m, j \cdot K_c + n}$, will be:

$$B_{i \cdot K_r + m, j \cdot K_c + n} = \frac{1}{K_r K_c} \{ (K_r - m)(K_c - n)A_{i+1,j+1} + m(K_c - n)A_{i+1,j+1} + n(K_r - m)A_{i+1,j+1} + nmA_{i+1,j+1} \} \quad (6.15)$$

$$\begin{aligned} \text{where } & 0 \leq i < R \\ & 0 \leq j < C \\ & 0 \leq m < K_r \\ & 0 \leq n < K_c \end{aligned}$$

Figure 6.13 shows an original image without using enhancement techniques. This image was obtained from a composite material with a hole in the center. Each pixel in the image is displayed as a box, therefore the resolution is not good. For an image using the enhancement technique as shown in Figure 6.14. The edges become smooth and an overall image quality improvement is obtained.

6.3.4 Tissue Characterization by Hierarchical Clustering

Once the information of each feature is obtained a pattern matrix can be constructed for further analysis. A pattern matrix is a matrix which contains the information of each feature in a column and all information obtained from a certain

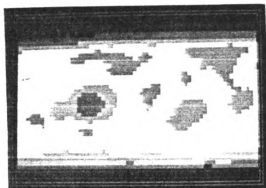


Figure 6.13. The original image obtained from a composite material with a hole in the center

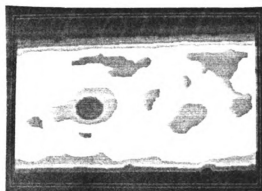


Figure 6.14. The enhanced image by interpolation

location in one row. In our experiments, we have 2601 rows, (51 by 51 locations), and 5 columns (5 different features). We can use an eigenvector projection to see the relationship between these five features. In the eigenvector projection, the two eigenvectors with the two most significant eigenvalues are chosen. The projection of five features in a two-dimensional space is shown in Figure 6.15. From Figure 6.15 we can observe that the power and cross correlation provide similar information, while the peak frequency and central frequency are close.

6.3.5 Data Reduction

Since the number of pixels in the images is large, it will take a long processing time and large memory space. The first step is to remove the informationless background of the images. This process will reduce the number of patterns necessary and thus simplify the problem with decreased processing time. The background of the images can easily be removed because it is very uniform. For the images obtained from the sample of human brain, the original image had 2601 pixels, while the image without background has only 1905 pixels. By these 1905 pixels a proximity matrix which is 1905 by 1905 will be constructed using the normalized pattern matrix with uncorrelated features. The proximity matrix will be symmetrical and the diagonal entries are all zero. Due to a 17% decrease in the number of pixels a 47% decrease in the size of the proximity matrix will result. Therefore the processing time will be greatly improved.

6.3.6 Image Reconstruction

Once the sequence of partitions is obtained, the pixels can be classified as numbers of groups desired. By giving each group of pixels a new label, new images can be constructed. The results of clustering by the single link, the complete link, Ward's

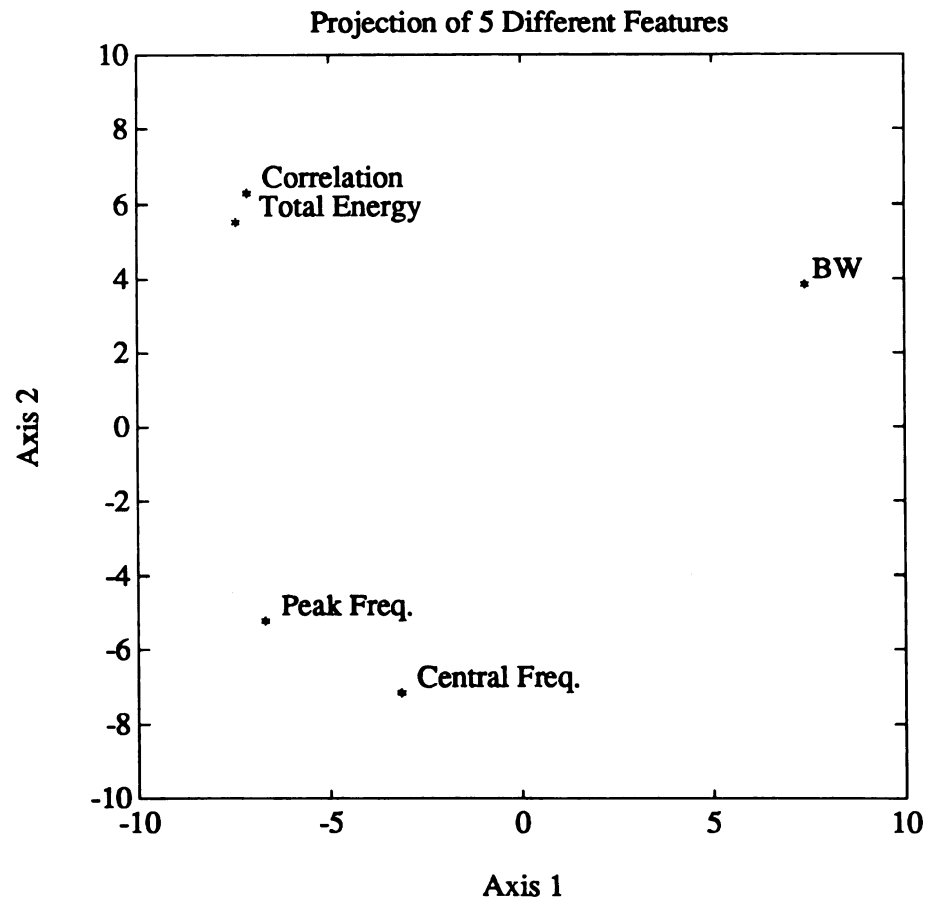


Figure 6.15. Projection of five features in a two-dimensional space

method, and the UPGMA are shown in Figures 6.16, 6.17, 6.18, and 6.19, respectively. In Figure 6.16, which is constructed by the single link method, the size of most clusters are small and separated, so it does not provide good information about the properties of tissue. We can only see that the upper part of the image is different from the other part of the tissue.

Figure 6.17 is constructed by the complete link method. This image is classified as 4 clusters including two clusters are mixed together along the edge of the sample. The distribution of these two clusters are close to the gray matter of the brain. This image looks better than the image obtained from the single link method.

The result of the Ward's method is shown in Figure 6.18. The gray matter and the area of tumor is clearly shown in the image, but some small clusters are distributed over the samples.

Finally the result of the UPGMA method is shown in Figure 6.19. The clustering result by this method looks pretty good. We can clearly see the gray matter, the white matter, and the hemorrhaged tumor which are all verses close to the original sample.

6.3.7 Cluster Validation

The discussion above focussed on direct observation for image validation. This may not be objective and is certainly not a quantitative way to evaluate the clustering results. In order to determine which method best fits the original data, the internal index, CPCC, of each clustering result is calculated. The value of the CPCC by different clustering methods for the section of human brain with a hemorrhaged tumor are shown in Table 6.2 and Figure 6.20.

From Table 6.2, the value of the CPCC by the single link method is only 0.07,



Figure 6.16. Clustering result of a section of human brain with a hemorrhaged tumor by single link clustering method with 8 clusters

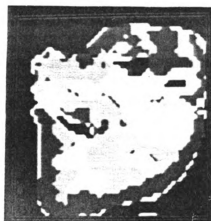


Figure 6.17. Clustering result of a section of human brain with a hemorrhaged tumor by complete link clustering method with 4 clusters



Figure 6.18. Clustering result of a section of human brain with a hemorrhaged tumor by Ward's clustering method with 9 clusters



Figure 6.19. Clustering result of a section of human brain with a hemorrhaged tumor by UPGMA clustering method with 3 clusters

which is very low. This means that the results obtained from the signal link in this experiment is not good but instead is close to random noise. Thus, we should not trust this clustering result.

The results obtained from the complete link and the Ward's method have higher values of CPCC. This implies that the result obtained by the complete link and by Ward's method are better than the result obtained by the single link method.

Comparing with the above three methods with the UPGMA method indicates that the latter method has the highest value of CPCC. Therefore, the results obtained from the UPGMA method should provide the best information about the original data. This is confirmed by direct observation of the images. We can see from Table 6.2 and the constructed image that the best result for our experiments is obtained by the UPGMA method.

Method	Value
Single Link	0.07
Complete Link	0.18
Ward's method	0.27
UPGMA	0.71

Table 6.2. The values of CPCC by different clustering methods for a section of human brain with a hemorrhaged tumor

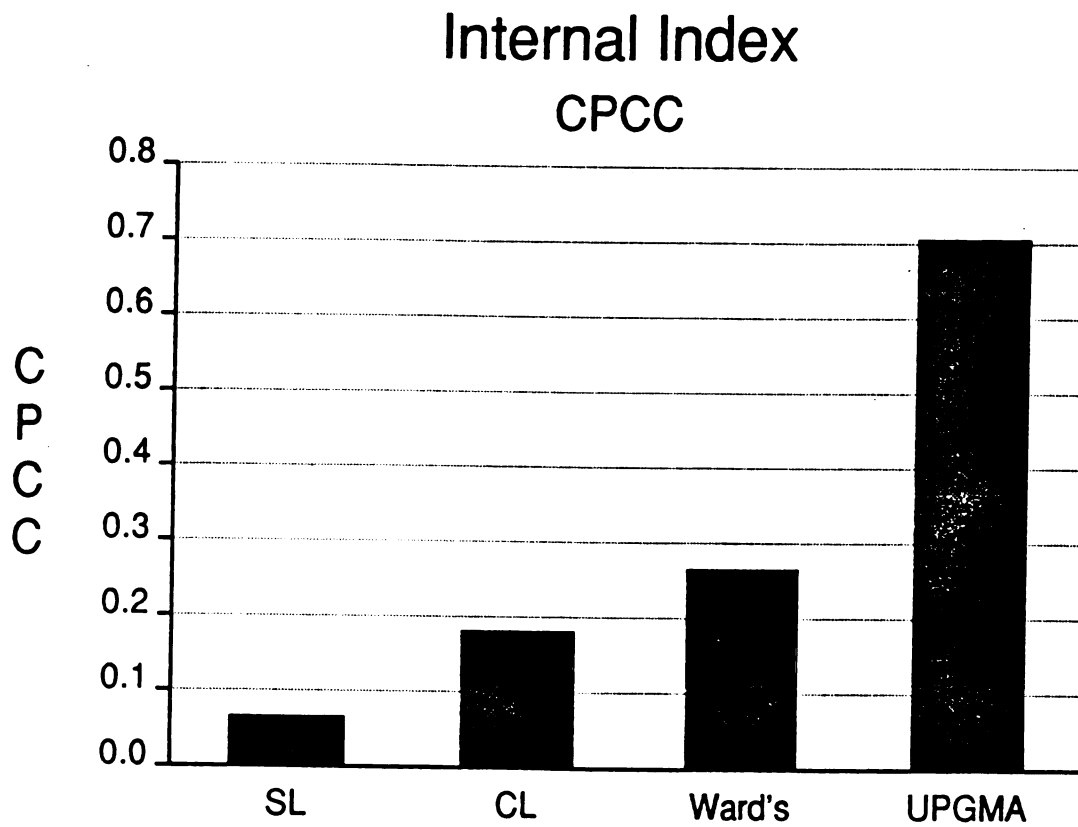


Figure 6.20. Comparison of CPCC by the different methods for a section of human brain with a hemorrhaged tumor

CHAPTER 7

CONCLUSIONS AND DIRECTIONS FOR FUTURE WORK

7.1 Conclusions

In this dissertation, some basic principles of acoustic waves have been reviewed and the techniques to find velocity and attenuation information by broadband signals have been derived and experimentally verified. Based on the unique spectral reflectance features of different materials, the velocity-density product and attenuation-density ratio can be obtained for multilayered models. From this information, material identification using the experimentally measured signal returns was achieved.

Acoustical images were obtained by broadband signals derived from narrow time video pulses. The images were characterized by hierarchical clustering techniques. As a result, the technique developed is promising for applications to nondestructive identification, evaluation of materials, and biological applications in general.

7.2 Directions for Future Work

Acoustical images are very important in nondestructive evaluation and biological applications. There are many topics to pursue in future research. For example, a three-dimensional acoustic image could be constructed with a processing computer which is sufficiently large and fast. Also a real time acoustical image system could be designed with digital signal processing chips. The accuracy of tissue characterization could be improved by building a database of the response from different biological tissue. Finally, using the technologies of artificial neural networks and artificial intelligence, an automatic diagnosis system could be constructed.

APPENDIX

APPENDIX A

PROGRAM LIST - SCANNING AND COLOR DISPLAY

```
/******  
*  
*      Scanning and Color Display:      *  
*  
*      Control motors X & Y to take data, find an echo in a certain *  
*      range, and show the image on screen.      *  
*  
*      By Nathan N. H. Wang      *  
*  
*****/  
  
#include <stdio.h>  
#include <stdlib.h>  
#include <math.h>  
#include <conio.h>  
#include <ctype.h>  
#include <graphics.h>  
  
#define VERSION "Color Scan V 9.3"  
#define AUTHOR "Nai-Hsien Wang"  
#define M_DATE "Apr. 29, 1991"  
  
/* #define DEBUGECHO      1*/  
/* #define DEBUGNOINP      1*/  
/* #define DEBUGPLOT      1*/  
  
#define SIZEX      61  
#define SIZEY      61
```

```

#define MAXSIZE          6600

#define MAXN              5
#define MAXERR            300
#define SIGTHRE           10
#define CHECKHT           0.2
#define NTRY              2

#define MOVE_POS          1
#define MOVE_NEG          -1
#define MOVEX             0
#define MOVEY             1
#define XSTEP             0.033333
#define YSTEP             0.033333

#define STANDARD          "INPUTX.DTA"
#define ECHOTEST          "ECHOTEST.DTA"
#define BEEP              '\007'
#define CTRLC             '\003'
#define ESC               27

unsigned int      sum[MAXSIZE];
signed int       sig[MAXSIZE];
signed int       c[SIZEX][SIZEY];
float            cht[SIZEX][SIZEY];
unsigned char    isum[MAXSIZE];

unsigned int      Israte;          /* index of sampling rate      */
unsigned int      Start;           /* starting point of sampling  */
unsigned int      Length;          /* length of sampling interval */
unsigned int      Trigger;         /* trigger level of sampling   */
unsigned int      Nsignal;         /* No. of signals to be averaged*/
unsigned int      Xmax;            /* max. step of motor X       */
unsigned int      Ymax;            /* max. step of motor Y       */
unsigned int      StepX;           /* scanning step of motor X   */
unsigned int      StepY;           /* scanning step of motor Y   */
unsigned int      Xdot;            /* display points in X axis   */
unsigned int      Ydot;            /* display points in Y axis   */
unsigned int      Xdot2;           /* distance between two points X*/
unsigned int      Ydot2;           /* distance between two points Y*/
unsigned char     Fhtabs;          /* flag to take abs or not    */
unsigned int      Delay;           /* delay time of stepper motors */
signed int        inpx[500];       /* input xt                    */
unsigned int      inpx_length;
unsigned int      pmax_inpx;
float             smean;
signed int        ismean;
long              xt2sum;

```

```

FILE          *fopdata;          /* pointer of output data file */
FILE          *fipdata;          /* pointer of input  data file */
char          fname[40];

int graphdriver;                  /* graphics card, eg. Hercules..*/
int graphmode;                    /* graphics mode                    */

int w0t,w0b,w0r,w0l,w0clip;      /* window 0: color bar            */
int w1t,w1b,w1r,w1l,w1clip;      /* window 1: image                 */
int w2t,w2b,w2r,w2l,w2clip;      /* window 2: command               */

int pcolor[9] = {RED,LIGHTRED,YELLOW,LIGHTGREEN,LIGHTCYAN,
                  CYAN,LIGHTBLUE,BLUE,BLACK};

int thre[11];                     /* normalized threshold            */
float htmin;                      /* real value of thre[0]           */
float htmax;                      /* real value of thre[10]          */
float htscale;                    /* scale of thre[]                 */
float htscale2;                   /* scale of thre[] for display     */
int x0,y0;                        /* origin of the image             */
int lx,ly;                        /* size of the image               */
float fthre[11];

void main()
{
    clrscr();
    printf("%s :\n%s, %s\n", VERSION, AUTHOR, M_DATE);
    mainfunc();
    fcloseall();
}                                  /* end of main()                    */

mainfunc()
{
    unsigned int  ansflag;
    unsigned char ans;
    unsigned char saveall;
    unsigned char datafile;
    int i;

    ansflag=0;
    thre[10]=0.0;
    htmin=0.0;
    while( ansflag == 0 )
    {
        printf("\n\n D: Display image from a file.\n");
        printf(" T: Take data from the scanning system.\n");
        printf(" F: Take data from a data file.\n");
        printf(" M: Get monochrome image by threshold.\n");
    }

```

```

printf("      Which one ? ");
ans=toupper(getch());
if ( ans == 'D' || ans == 'T' || ans == 'F' || ans == 'M') ansflag=1;
else printf("%c",BEEP);
}
printf("\n\n");
switch (ans)
{
    case 'D':
        Xdot=4;
        Ydot=4;
        Xdot2=4;
        Ydot2=4;
        loading();
        getcommand();
        break;

    case 'T':
        loadxt();
        for(i=0;i<10;i++)
            thre[i]=(int)((9-i)*256/9.0+.5);
        initmotor();
        resetmotor();
        inputdata(&saveall,&datafile);
        scandata(saveall,datafile);
        getcommand();
        break;

    case 'F':
        readdata();
        getcommand();
    case 'M':
        setthre();
        getcommand();
}
/* end of switch */
}
/* end of mainfunc() */

readdata()
{
    int          i;
    char          fname1[40];
    char          tmp;
    unsigned char datafile;

    loadxt();
    for(i=0;i<10;i++) thre[i]=(int)((9-i)*256/9.0+.5);
    printf("\nData filename =?");
    scanf("%s", fname1);

```

```

fipdata = fopen(fname1, "rb");
if (fipdata <= 0)
{
    printf("\n %c Cannot open file: %s.\n",BEEP,fname1);
    return (-1);      /* cannot open file, return */
}
fscanf(fipdata,"%d %d %d %d %d %d %d %d ",&Israte,&Xmax,&Ymax
        ,&StepX,&StepY,&Start,&Length,&Nsignal);
fscanf(fipdata,"%d %d %d %d %d %d %g",&Xdot,&Ydot,&Xdot2
        ,&Ydot2,&Trigger,&Fhtabs,&htmax);
fscanf(fipdata,"%c",&tmp);          /* delete LF    */
if (Fhtabs >90) datafile='S';
Fhtabs=toupper(Fhtabs);
scandata2(datafile);
}                                     /* end of readdata()      */

setthre()
{
    int          i;
    char          fname1[40];
    char          tmp;
    unsigned char datafile;

    for(i=0;i<10;i++) thre[i]=(int)((9-i)*256/9.0+.5);
    printf("\nData filename =?");
    scanf("%s", fname1);
    fipdata = fopen(fname1, "rb");
    if (fipdata <= 0)
    {
        printf("\n %c Cannot open file: %s.\n",BEEP,fname1);
        return (-1);      /* cannot open file, return */
    }
    fscanf(fipdata,"%d %d %d %d %d %d %d %d ",&Israte,&Xmax,&Ymax
            ,&StepX,&StepY,&Start,&Length,&Nsignal);
    fscanf(fipdata,"%d %d %d %d %d %d %e",&Xdot,&Ydot,&Xdot2
            ,&Ydot2,&Trigger,&Fhtabs,&htmax);
    fscanf(fipdata,"%c",&tmp);          /* delete LF    */
    if (Fhtabs >90) datafile='S';
    Fhtabs=toupper(Fhtabs);
    scandata3(datafile);
}                                     /* end of setthre()      */

loading()                             /* load image from a file */
{
    FILE          *fipimg;            /* pointer of input image file */
    int           i,j;
    char          fname2[40];

```

```

closegraph();
printf("Input image data filename = ? ");
scanf("%s",fname2);
fipimg=fopen(fname2,"r");
if (fipimg <= 0)
{
    printf("\n %c Cannot open file: %s.\n",BEEP,fname2);
    printf("\n Press any key to continue.\n");
    getch();
}
else
{
    strcpy(fname,fname2);
    fscanf(fipimg,"%d %d %d %d %d %f %f",&Xmax,&Ymax,&StepX,&StepY,
        &Fhtabs,&htmax,&htmin);
    fscanf(fipimg,"%d %d %d %d %d",&Israte,&Start,&Length,
        &Trigger,&Nsignal);
    for (i=0;i<9;i++)
    {
        fscanf(fipimg,"%d",&pcolor[i]);
    }
    for (i=0;i<10;i++)
    {
        fscanf(fipimg,"%d",&thre[i]);
    }
    for (j=0;j<=Ymax;j++)
    {
        for (i=0;i<=Xmax;i++)
        {
            fscanf(fipimg,"%d %e",&c[i][j],&cht[i][j]);
        }
    }
    fclose(fipimg);
}
showall();
}                                     /* end of loading()          */

saveimg()                             /* save image to a file      */
{
    FILE          *fopimg;           /* pointer of output image */
    int            i,j;
    char           ans;
    char           fname2[40];

    closegraph();
    printf("Output image data filename = ? ");
    scanf("%s",fname2);
    if (access(fname2,0) == 0)

```

```

{
    printf("\nFile [%s] exists, Replace (Y or N) ? ",fname2);
    do {
        ans=toupper(getch());
        if (ans == 'Y') printf("\nFile will be replaced.\n");
        if (ans == 'N')
        {
            printf("\nFile not saved, press any key to return.\n");
            getch();
            showall();
            return(-1);
        }
    } while(ans != 'Y' && ans != 'N');
}
fopimg=fopen(fname2,"w");
if (fopimg <= 0)
{
    printf("\n %c Cannot open file: %s.\n",BEEP,fname2);
    printf("\n Press any key to continue.\n");
    getch();
}
else
{
    strcpy(fname,fname2);
    fprintf(fopimg,"%5d %5d %5d %5d %5d %g %g\n",Xmax,Ymax,StepX,StepY,
        Fhtabs,htmax,htmin);
    fprintf(fopimg,"%5d %5d %5d %5d %5d\n",Israte,Start,Length,
        Trigger,Nsignal);
    for (i=0;i<9;i++)
    {
        fprintf(fopimg,"%5d ",pcolor[i]);
    }
    fprintf(fopimg,"\n");
    for (i=0;i<10;i++)
    {
        fprintf(fopimg,"%5d ",thre[i]);
    }
    fprintf(fopimg,"\n");
    for (j=0;j<=Ymax;j++)
    {
        for (i=0;i<=Xmax;i++)
        {
            fprintf(fopimg,"%6d %6.3f \n",c[i][j],cht[i][j]);
        }
    }
    fclose(fopimg);
}
showall();

```

```

}                                     /* end of saveimg()          */

loadxt()                             /* load x(t) from INPUTX.DTA    */
{

    FILE          *fip;
    unsigned int   i;
    float          inpxf;
    float          samplerate;      /* samplerate, israte, startpoint, */
    unsigned int   israte;          /* , and j1 are not used in        */
    unsigned int   startpoint;      /* this subroutine. They are set   */
    unsigned int   j1;              /* to be compatible with the      */
                                    /* data file which are generated   */
                                    /* by AD?.c.                       */

    float          stest;
    FILE          *ftest;

    fip = fopen(STANDARD,"r");
    if (fip<=0)                      /* Cannot open file            */
    {
        printf("\n\n%cCannot find standard input file %s !\n\n",
            BEEP,STANDARD);
        return(-1);
    }
    printf("\nRead standard signal from %s.",STANDARD);

    fscanf(fip,"%f %d %d %d %f ",
        &samplerate,&israte,&startpoint,&j1,&smean);

    for (i = 0; !feof(fip); i++)
    {
        fscanf(fip, "%f ", &inpxf);
        inpx[i]=inpxf-smean+0.5;
    }
    fclose(fip);

    inpx_length=i-1;
    pmax_inpx=0;
    xt2sum=0;
    for (i=0; i<inpx_length; i++)
    {
        if (abs(inpx[i]) > abs(inpx[pmax_inpx])) pmax_inpx=i;
        xt2sum=xt2sum+inpx[i]*inpx[i];
    }
    /* end of for */
#ifdef DEBUGECHO
    ftest=fopen(ECHOTEST,"r");
    printf("\nRead data from ECHOTEST.DTA. ");
    for (i = 0; !feof(ftest); ++i)

```

```

    {
        fscanf(ftest, "%f", &stest);
        sig[i]=(unsigned int)(-(stest-smean)*0.75+smean);
        printf("%d ",sig[i]);
    }
    fclose(ftest);
    Length=i-1;
    Start=0;
#endif
    ismean=(int)(smean+0.5);
} /* end of loadxt() */

inputdata(unsigned char *psave,unsigned char *pdatafile)
/* get setup parameters */
{
    unsigned char saveall;
    unsigned int loopflag;
    char         fname1[40]; /* filename of output file */
    float        xmax1;
    float        ymax1;
    float        xstep1;
    float        ystep1;
    char         ans;

    do {
        printf("\nChoose the sampling rate:\n");
        printf("\n      0: 20MHz      8: 40MHz");
        printf("\n      1: 2MHz       9: 4MHz");
        printf("\n      2: 200KHz     10: 400KHz");
        printf("\n      3: 20KHz      11: 40KHz");
        printf("\n      4: 2KHz       12: 4KHz");
        printf("\n      Sampling rate = ?");
        scanf("%d",&Israte);
    } while (Israte<0 || Israte>12 || ( Israte>4 && Israte<8));
#ifdef DEBUGECHO
    printf("\n\nStarting point of sampling = ?");
    scanf("%d",&Start);
    printf("\nLength of sampling points (max no. = %d) = ?",MAXSIZE);
    scanf("%d",&Length);
    printf("\nTrigger signal offset (0-255, 128 = 0 volt) = ?");
    scanf("%d",&Trigger);
#endif
    Fhtabs='Y';
    printf("No. of signals to be averaged = ?");
    scanf("%d",&Nsignal);
    do
    {
        loopflag=0;

```

```

printf("\nMax scanning range in X direction (in mm) = ?");
scanf("%f",&xmax1);
printf("Max scanning range in Y direction (in mm) = ?");
scanf("%f",&ymax1);
printf("\nScanning step in X direction (in mm) = ?");
scanf("%f",&xstep1);
printf("Scanning step in Y direction (in mm) = ?");
scanf("%f",&ystep1);
Xmax=(int)(xmax1/xstep1);
Ymax=(int)(ymax1/ystep1);
StepX=(int)(xstep1/XSTEP);
StepY=(int)(ystep1/YSTEP);
printf("\nXmax = %6d, xstep = %6d,
        XSTEP = %f",Xmax,StepX,StepX*XSTEP);
printf("\nYmax = %6d, ystep = %6d,
        YSTEP = %f\n",Ymax,StepY,StepY*YSTEP);
if (Xmax>SIZEX)
{
    printf("\nToo many points in X, ",
           "max. points in X is %d, try again\n",
           SIZEX);
    loopflag=1;
}
if (Ymax>SIZEY)
{
    printf("\nToo many points in Y, ",
           "max. points in Y is %d, try again\n",
           SIZEY);
    loopflag=1;
}
} while (loopflag == 1);

printf("\nDisplay size of each pixel in X axis = ?");
scanf("%d",&Xdot);
printf("Display size of each pixel in Y axis = ?");
scanf("%d",&Ydot);
Xdot2=Xdot;
Ydot2=Ydot;
printf("\nMax. ht = ?");
scanf("%f",&htmax);
printf("Delay time (msec) of stepper motors (>= 3) = ?");
scanf("%d",&Delay);
do {
    printf("\nSave all sampling points (Y or N) = ?");
    saveall=getch();
    printf("%c\n",saveall);
    *psave=toupper(saveall);
    ans='Y';

```

```

if (*psave == 'Y')
{
    printf("\nSmall size of Large size data file (S or L) = ?");
    *pdatafile=toupper(getch());
    if (*pdatafile == 'S') printf("Small data file.\n");
    else                    printf("Large data file.\n");
    printf("\nOutput filename =?");
    scanf("%s", fname1);
    if (access(fname1,0) == 0)
    {
        printf("\nFile [%s] exists, Replace (Y or N) ? ",fname1);
        ans=toupper(getch());
        if (ans == 'Y') printf("\nFile will be replaced.\n");
        if (ans == 'N') printf("\nNot replaced\n");
    } while(ans != 'Y' && ans != 'N');
    if (ans == 'Y')
    {
        fopdata = fopen(fname1, "wb");
        if (fopdata <= 0)
        {
            printf("\n %c Cannot open file: %s.\n",BEEP,fname1);
            return (-1);      /* cannot open file, return */
        }
        if (*pdatafile == 'S') Fhtabs=tolower(Fhtabs);
            /* if Fhtabs is captial, then the data */
            /* file is large, otherwise is small */

        fprintf(fopdata,"%5d %5d %5d %5d %5d %5d %5d %5d\n",
            Israte,Xmax,Ymax,StepX,StepY,Start,Length,Nsignal);
        fprintf(fopdata,"%5d %5d %5d %5d %5d %5d %11.4e\n",
            Xdot,Ydot,Xdot2,Ydot2,Trigger,Fhtabs,htmax);
        Fhtabs=toupper(Fhtabs);
    }
} while(ans == 'N');
} /* end of inputdata() */

scandata(saveall,datafile) /* control motors and get data */
unsigned char  saveall;
unsigned char  datafile;
{
    signed int  xp;
    signed int  yp;
    unsigned int  i,j;
    float        err;
    unsigned int  goodsig;
    unsigned int  ntotal;
    unsigned int  echo;

```

```

float      echoht;
int        try;
int        tryflag;

detectgraph(&graphdriver,&graphmode);
initplot();

for (yp=0; yp <= Ymax; yp+=2) {
    #ifndef DEBUGECHO
        try=0;
        do
        {
            try++;
            tryflag=0;
            takesample(&echo,&echoht,&goodsig,&ntotal);
                                                    /* sampling at (0,yp)          */
            if ( (yp>0) && (try <NTRY) )
            {
                if ((fabs((double)(echoht-cht[0][yp-1])) > CHECKHT) &&
                    (fabs((double)(echoht-cht[1][yp])) > CHECKHT))
                    tryflag=1;
            }
        } while ( (tryflag==1) && (try<NTRY));
    #else
        findiecho(sig,&echo,&echoht);
    #endif

    movemotor(MOVEX,MOVE_POS,StepX,Delay);
    if (saveall == 'Y') saveallp(0,yp,sum,goodsig,ntotal,datafile);
    c[0][yp]=echo;
    cht[0][yp]=echoht;
    display(0,yp);

    for (xp=1; xp <= Xmax; xp++)
    {
        #ifndef DEBUGECHO
            try=0;
            do
            {
                try++;
                tryflag=0;
                takesample(&echo,&echoht,&goodsig,&ntotal);
                                                    /* sampling at (xp,yp) */
                if ( try <NTRY )
                {
                    if (yp >0)
                    {
                        if ((fabs((double)(echoht-cht[xp][yp-1])) > CHECKHT) &&

```

```

        (fabs((double)(echoht-cht[xp-1][yp])) > CHECKHT))
        tryflag=1;
    }
    else
        if (fabs((double)(echoht-cht[xp-1][yp])) > CHECKHT)
            tryflag=1;
    }
    } while ( (tryflag==1) && (try<NTRY));
#else
    findiecho(sig,&echo,&echoht);
#endif
if ( xp < Xmax ) movemotor(MOVEX,MOVE_POS,StepX,Delay);
    else          movemotor(MOVEY,MOVE_POS,StepY,Delay);
if (saveall == 'Y') saveallp(xp,yp,sum,goodsig,ntotal,datafile);
c[xp][yp]=echo;
cht[xp][yp]=echoht;
display(xp,yp);
}

if ( yp < Ymax )
{
    #ifndef DEBUGECHO
        try=0;
        do
        {
            try++;
            tryflag=0;
            takesample(&echo,&echoht,&goodsig,&ntotal);
                                /* sampling at (Xmax,yp+1) */
            if ( try < NTRY )
            {
                if (fabs((double)(echoht-cht[Xmax][yp])) > CHECKHT)
                    tryflag=1;
            }
        } while ( (tryflag==1) && (try<NTRY));
    #else
        findiecho(sig,&echo,&echoht);
    #endif
    movemotor(MOVEX,MOVE_NEG,StepX,Delay);
    if (saveall == 'Y')
        saveallp(Xmax,yp+1,sum,goodsig,ntotal,datafile);
    c[Xmax][yp+1]=echo;
    cht[Xmax][yp+1]=echoht;
    display(Xmax,yp+1);

    for (xp=Xmax-1; xp >= 0; xp--)
    {
        #ifndef DEBUGECHO

```

```

try=0;
do
{
try++;
tryflag=0;
takesample(&echo,&echoht,&goodsig,&ntotal);
/* sampling at (xp,yp+1) */

if ( try < NTRY )
{
if ((fabs((double)(echoht-cht[xp][yp])) > CHECKHT) &&
(fabs((double)(echoht-cht[xp+1][yp+1])) > CHECKHT))
tryflag=1;
}
} while ( (tryflag==1) && (try<NTRY));
#else
findiecho(sig,&echo,&echoht);
#endif

if ( xp > 0 ) movemotor(MOVEX,MOVE_NEG,StepX,Delay);
else movemotor(MOVEY,MOVE_POS,StepY,Delay);
if (saveall == 'Y')
saveallp(xp,yp+1,sum,goodsig,ntotal,datafile);
c[xp][yp+1]=echo;
cht[xp][yp+1]=echoht;
display(xp,yp+1);
} /* end of for xp */
} /* end of if (yp < Ymax) */
} /* end of for loop (yp) */
if (saveall == 'Y') fclose(fopdata);
} /* end of scandata() */

scandata2(datafile) /* readdata from file */
unsigned char datafile;
{
signed int xp;
signed int yp;
unsigned int i,j;
float err;
unsigned int goodsig;
unsigned int ntotal;
unsigned int echo;
float echoht;

detectgraph(&graphdriver,&graphmode);
initplot();

for (yp=0; yp <= Ymax; yp+=2) {
loadallp(datafile);

```

```

findiecho(sig,&echo,&echoht);    /* echo at (0,yp)          */
c[0][yp]=echo;
cht[0][yp]=echoht;
display(0,yp);

for (xp=1; xp <= Xmax; xp++)
{
    loadallp(datafile);
    findiecho(sig,&echo,&echoht); /* echo at (xp,yp)          */
    c[xp][yp]=echo;
    cht[xp][yp]=echoht;
    display(xp,yp);
}

if ( yp < Ymax )
{
    loadallp(datafile);
    findiecho(sig,&echo,&echoht); /* echo at (Xmax,yp+1)      */
    c[Xmax][yp+1]=echo;
    cht[Xmax][yp+1]=echoht;
    display(Xmax,yp+1);

    for (xp=Xmax-1; xp >= 0; xp--)
    {
        loadallp(datafile);
        findiecho(sig,&echo,&echoht);    /* echo at (xp,yp+1)      */
        c[xp][yp+1]=echo;
        cht[xp][yp+1]=echoht;
        display(xp,yp+1);
    }
}
/* end of for xp          */
/* end of if (yp < Ymax)  */
/* end of for loop (yp)   */
/* end of scandata2()     */

scandata3(datafile)    /* readdata from file      */
unsigned char    datafile;
{
    signed int    xp;
    signed int    yp;
    unsigned int  i,j;
    float         err;
    unsigned int  goodsig;
    unsigned int  ntotal;
    unsigned int  echo;
    float         echoht;
    unsigned int  flag1;
    unsigned int  flag2;
    int           stemp;

```

```

float      upthre;
float      lowthre;
int        iupthre;
int        ilowthre;

printf("\nInput the range of threshold.\n");
printf("Upper bound of the signal (0.0 - 1.0) = ?");
scanf("%f",&upthre);
printf("Lower bound of the signal (0.0 - %f) = ?",upthre);
scanf("%f",&lowthre);
iupthre=upthre*128;
ilowthre=lowthre*128;

detectgraph(&graphdriver,&graphmode);
initplot();

for (yp=0; yp <= Ymax; yp+=2) {
    loadallp(datafile);
    flag1=0;
    flag2=0;
    for (i=0;i<Length;i++)
    {
        stemp=abs(sig[i]-128);
        if (stemp >= ilowthre) flag1=1;
        if (stemp >= iupthre) flag2=1;
    }
    if ((flag1 == 1) && (flag2 == 0))
    {
        c[0][yp]=1;
        cht[0][yp]=lowthre*htmax;
    }
    else
    {
        c[0][yp]=0;
        cht[0][yp]=0;
    }
    display(0,yp);

    for (xp=1; xp <= Xmax; xp++)
    {
        loadallp(datafile);
        flag1=0;
        flag2=0;
        for (i=0;i<Length;i++)
        {
            stemp=abs(sig[i]-128);
            if (stemp >= ilowthre) flag1=1;
            if (stemp >= iupthre) flag2=1;

```

```

    }
    if ((flag1 == 1) && (flag2 == 0))
    {
        c[xp][yp]=1;
        cht[xp][yp]=lowthre*htmax;
    }
    else
    {
        c[xp][yp]=0;
        cht[xp][yp]=0;
    }
    display(xp,yp);
}

if ( yp < Ymax )
{
    loadallp(datafile);
    flag1=0;
    flag2=0;
    for (i=0;i<Length;i++)
    {
        stemp=abs(sig[i]-128);
        if (stemp >= ilowthre) flag1=1;
        if (stemp >= iupthre) flag2=1;
    }
    if ((flag1 == 1) && (flag2 == 0))
    {
        c[Xmax][yp+1]=1;
        cht[Xmax][yp+1]=lowthre*htmax;
    }
    else
    {
        c[Xmax][yp+1]=0;
        cht[Xmax][yp+1]=0;
    }
    display(Xmax,yp+1);

    for (xp=Xmax-1; xp >= 0; xp--)
    {
        loadallp(datafile);
        flag1=0;
        flag2=0;
        for (i=0;i<Length;i++)
        {
            stemp=abs(sig[i]-128);
            if (stemp >= ilowthre) flag1=1;
            if (stemp >= iupthre) flag2=1;
        }
    }
}

```

```

        if ((flag1 == 1) && (flag2 == 0))
        {
            c[xp][yp+1]=1;
            cht[xp][yp+1]=lowthre*htmax;
        }
        else
        {
            c[xp][yp+1]=0;
            cht[xp][yp+1]=0;
        }
        display(xp,yp+1);
    }
}
/* end of for xp */
/* end of if (yp < Ymax) */
/* end of for loop (yp) */
/* end of scandata3() */

saveallp(xp,yp,sum,goodsig,ntotal,datafile)
signed int    xp;
signed int    yp;
unsigned int   sum[];
unsigned int   goodsig;
unsigned int   ntotal;
unsigned char  datafile;
{
    unsigned int  i;

    fprintf(fopdata,"%c%c%c%c",xp,yp,goodsig,ntotal);
    for (i=0;i<Length;i++) sig[i]+=ismean;

    if (datafile == 'S')
    {
        if (goodsig ==1)
            for (i=0;i<Length;i++) isum[i]=sig[i];
        else
            for (i=0;i<Length;i++) isum[i]=(sum[i]+goodsig/2.0)/goodsig;
        fwrite(isum,sizeof(char),Length,fopdata);
    }
    else
    {
        if (goodsig ==1)
            fwrite(sig,sizeof(int),Length,fopdata);
        else
            fwrite(sum,sizeof(int),Length,fopdata);
    }
}

loadallp(datafile)
unsigned char  datafile;
{

```

```

unsigned int  i;
unsigned int  xp;
unsigned int  yp;
unsigned int  goodsig;
unsigned int  ntotal;
unsigned char tmp;

fscanf(fipdata,"%c",&tmp);          /* read xp      */
fscanf(fipdata,"%c",&tmp);          /* read yp      */
fscanf(fipdata,"%c",&tmp);          /* read goosig  */
goodsig=tmp;
fscanf(fipdata,"%c",&tmp);          /* read ntotal  */

if (datafile == 'S')
{
    fread(isum,sizeof(char),Length,fipdata);
    for (i=0;i<Length;i++) sig[i]=isum[i];
}
else
{
    fread(sig,sizeof(int),Length,fipdata);
    for (i=0;i<Length;i++)
        sig[i]=(sig[i]+goodsig/2)/goodsig;
}
}

initplot()
{
    unsigned int  maxx;
    unsigned int  maxy;
    int           i,j,k;
    int           x,y;
    float         xmaxl;
    float         ymaxl;
    float         xstepl;
    float         ystepl;
    int           iscale;

    initgraph(&graphdriver,&graphmode,"\\tc\\");
    setpalette(5,36);
    setpalette(13,32);
    maxx=getmaxx();
    maxy=getmaxy();

    w0l = maxx*3/4;          /* parameters for window 0 */
    w0t = 50;                /* for color bar.          */
    w0r = maxx;
    w0b = (maxy*3)/4;

```

```

w0clip =1;

w1l = 10;           /* parameters for window 1    */
w1t = 10;           /* for image.                                     */
w1r = w0l-1;
w1b = w0b;
w1clip =1;

w2l = 0;             /* parameters for window 2    */
w2t = w0b+20;        /* for command line.          */
w2r = maxx;
w2b = maxy;
w2clip =1;

if(Fhtabs == 'Y')
    htscale=(htmax-htmin)/256.0;
else
{
    htmin=-htmax;
    htscale=htmax/128.0;
}
setviewport(w0l,w0t,w0r,w0b,w0clip);
for ( i=0;i<9;i++)
{
    k=60+i*20;
    setfillstyle(SOLID_FILL,pcolor[i]);
    bar(3,k,52,k+19);
}
setcolor(BROWN);
rectangle(1,59,54,241);

x=60;
setcolor(WHITE);
iscale=0;
htscale2=1.0;

while (htmax > 10.0/htscale2)
{
    htscale2*=0.1;
    iscale++;
}
while (htmax < 1.0/htscale2)
{
    htscale2*=10.0;
    iscale--;
}
if (iscale !=0)
{

```

```

        y=20;
        x=60;
        gprintf(&x,&y,"      %1d",iscale);
        gprintf(&x,&y," x 10");
    }

for (i=0;i<10;i++)
{
    y=56+i*20;
    fthre[i]=htscale*thre[i]+htmin;
    gprintf(&x,&y,"%6.3f",fthre[i]*htscale2);
}
fthre[10]=htscale*thre[10]+htmin;

xstep1=StepX*XSTEP;
ystep1=StepY*YSTEP;
xmax1=Xmax*xstep1;
ymax1=Ymax*ystep1;

x=1;
y=280;
gprintf(&x,&y,"Size:%4.1fmm x %4.1fmm",xmax1,ymax1);
gprintf(&x,&y,"Res.:%4.1fmm x %4.1fmm",xstep1,ystep1);

setviewport(w1l,w1t,w1r,w1b,w1clip);
lx=Xdot2*(Xmax+1);
ly=Ydot2*(Ymax+1);
x0=w1r-((w1r-w1l-lx)/2);
y0=w1b-((w1b-w1t-ly)/2);

setcolor(BROWN);
rectangle(x0+2,y0+2,x0-lx-2,y0-ly-2);
rectangle(x0+6,y0+6,x0-lx-6,y0-ly-6);
}                                     /* end of initplot()          */

getcommand()
{
#define MES1      "Use arrow keys to change threshold."
#define MES2      " I:  toggle instant update."
#define MES3      "<CR>: return to main menu."
    int    x;
    int    y;
    int    flag;
    char    ans;
    int    i;
    char    ans2;
    char    ans3;

```

```

int    oldth;
int    ith;
int    flag2;
unsigned char saveall;
unsigned char datafile;
int    dispflag;

clearcursor();
dispcmd(WHITE);
flag=1;

while (flag ==1)
{
    do {
    } while (kbhit() ==0 );
    ans=toupper(getch());
    switch (ans)
    {
        case 'L':
            loading();
            break;
        case 'S':
            saveing();
            break;
        case 'T':
            dispcmd(BLACK);
            x=150;
            y=10;
            setviewport(w2l,w2t,w2r,w2b,w2clip);
            setcolor(WHITE);
            gprintf(&x,&y,"%s",MES1);
            gprintf(&x,&y,"%s",MES2);
            gprintf(&x,&y,"%s",MES3);
            x=60;
            ith=1;
            setviewport(w0l,w0t,w0r,w0b,w0clip);
            setcolor(GREEN);
            y=56+ith*20;
            gprintf(&x,&y,"%6.3f", (htscale*thre[ith]+htmin)*htscale2);
            flag2=1;
            dispflag=1;
            while(flag2==1)
            {
                setviewport(w0l,w0t,w0r,w0b,w0clip);
                do {
                } while (kbhit() ==0 );
                ans2=getch();
                if (ans2 == '\r')

```

```

{
    flag2=0;
    setcolor(WHITE);
    y=56+ith*20;
    gprintf(&x,&y,"%6.3f",(htscale*thre[ith]+htmin)*htscale2);
    x=150;
    y=10;
    setviewport(w2l,w2t,w2r,w2b,w2clip);
    setcolor(BLACK);
    gprintf(&x,&y,"%s",MES1);
    gprintf(&x,&y,"%s",MES2);
    gprintf(&x,&y,"%s",MES3);
    dispimg();
    dispcmd(WHITE);
}
else
{
    if (ans2 == '\0')
    {
        ans3=getch();
        switch(ans3)
        {
            case 'P':                                /* down          */
                setcolor(WHITE);
                y=56+ith*20;
                gprintf(&x,&y,"%6.3f",
                    (htscale*thre[ith]+htmin)*htscale2);
                if (ith < 8 ) ith++;
                setcolor(GREEN);
                y=56+ith*20;
                gprintf(&x,&y,"%6.3f",
                    (htscale*thre[ith]+htmin)*htscale2);
                break;
            case 'H':                                /* up              */
                setcolor(WHITE);
                y=56+ith*20;
                gprintf(&x,&y,"%6.3f",
                    (htscale*thre[ith]+htmin)*htscale2);
                if (ith > 1 ) ith--;
                setcolor(GREEN);
                y=56+ith*20;
                gprintf(&x,&y,"%6.3f",
                    (htscale*thre[ith]+htmin)*htscale2);
                break;
            case 'K':                                /* left           */
                oldth=thre[ith];
                if (thre[ith] > thre[ith+1]) thre[ith]--;
                setcolor(BLACK);

```



```

        closegraph();
        printf("\nMax. ht (current = %f) = ? ",htmax);
        scanf("%f",&htmax);
        printf("\nMin. ht (current = %f) = ? ",htmin);
        scanf("%f",&htmin);
        showall();
        break;
    case ' ':
        dispcmd(BLACK);
        do {
            } while (kbhit() ==0 );
        getch();
        dispcmd(WHITE);
        break;
    case 'X':
        changecolor();
        break;
    case 'N':
        closegraph();
        printf("%s :\n%s,  %s\n", VERSION, AUTHOR, M_DATE);
        loadxt();
        resetmotor();
        inputdata(&saveall,&datafile);
        scandata(saveall,datafile);
        clearcursor();
        dispcmd(WHITE);
        break;
    case ESC:
        flag=0;
        closegraph();
        break;
    default:
        flag=1;
}
/* end of switch() */
}
/* end of while */
}
/* end of getcommand() */

display(signed int xp, signed int yp)
{
    int    x1;
    int    y1;
    int    i;
    float  ht;
    int    ith;

    x1=x0-Xdot2*xp;
    y1=y0-Ydot2*yp;
    setfillstyle(SOLID_FILL,BLACK);

```

```

if ((xp == 0) || (xp == Xmax))
{
    bar(x0+3,y0-ly-1,x0+5,y0);          /* clear cursor in Y axis      */
    bar(x0-lx-5,y0-ly-1,x0-lx-3,y0);
    bar(x0-lx,y0+3,x0,y0+4);          /* clear cursor in X axis    */
    bar(x0-lx,y0-ly-4,x0,y0-ly-3);
    setfillstyle(SOLID_FILL,WHITE);
    bar(x0+3,y1-Ydot+1,x0+5,y1);      /* set cursor in Y axis      */
    bar(x0-lx-5,y1-Ydot+1,x0-lx-3,y1);
}
else
{
    bar(x1-Xdot-Xdot+1,y0+3,x1+Xdot,y0+4); /* clear cursor in X axis */
    bar(x1-Xdot-Xdot+1,y0-ly-4,x1+Xdot,y0-ly-3);
}

setfillstyle(SOLID_FILL,WHITE);
bar(x1-Xdot+1,y0+3,x1,y0+4);          /* set cursor in X axis      */
bar(x1-Xdot+1,y0-ly-4,x1,y0-ly-3);

#ifdef DEBUGPLOT
    cht[xp][yp]=(float)(xp+yp)/(Xmax+Ymax)*3.0-1.5;
#endif
ht=cht[xp][yp];
if(Fhtabs == 'Y') ht=fabs((double)ht);
ith=-1;
do
{
    ith++;
} while (( (ht < fthre[ith]) && (ht < fthre[ith+1]) ) && (ith < 8) );

setfillstyle(SOLID_FILL,pcolor[ith]);
bar(x1-Xdot+1,y1-Ydot+1,x1,y1);
if (kbhit() != 0)
{
    if (getch() == CTRLC)
    {
        closegraph();
        fcloseall();
        exit(-1);
    }
}
}

find1echo(sig,pecho,pechoht)
signed int    sig[];
unsigned int   *pecho;
float          *pechoht;

```

```

{
    long          ytsun;
    long          ytsummax;
    int           i,j;
    unsigned int   echo;
    float         ht;

    for (i=0;i<Length;i++) sig[i]-=ismean;
    echo=0;
    ht=0.0;
    if (Length>inpx_length)
    {
        ytsummax=0;
        for (i=0;i<Length-inpx_length;i++)
        {
            if (abs(sig[i+pmax_inpx]) > SIGTHRE )
            {
                ytsun=0;
                for (j=0;j<inpx_length;j++) ytsum=ytsum+sig[i+j]*inpx[j];
                if (labs(ytsum) > labs(ytsummax))
                {
                    ytsummax=ytsum;
                    echo=i+pmax_inpx+Start;
                }
            }
        }
        ht=(float)ytsummax/xt2sum;
    }
    else
    {
        echo=0;
        ht=0.0;
    }
    *pecho=echo;
    *pechoht=ht;
    #ifdef DEBUGECHO
        for (i=0;i<Length;i++) sig[i]+=ismean;
    #endif
}

dispimg()
{
    signed int     xp;
    signed int     yp;
    signed int     ith;
    float          ht;
    signed int     x1;
    signed int     y1;

```

```

signed int    i;
int           x;
int           y;

setviewport(w1l,w1t,w1r,w1b,w1clip);
setcolor(WHITE);
x=x0-lx;
y=w1b-w1t-20;
gprintf(&x,&y,"File = [%s]",fname);
for (i=0;i<11;i++) fthre[i]=htscale*thre[i]+htmin;
for (yp=0;yp<= Ymax;yp++)
{
    y1=y0-Ydot2*yp;
    setfillstyle(SOLID_FILL,BLACK);
    bar(x0+3,y0-ly-1,x0+5,y0);
    bar(x0-lx-5,y0-ly-1,x0-lx-3,y0);

    setfillstyle(SOLID_FILL,WHITE);
    bar(x0+3,y1-Ydot-1,x0+5,y1);
    bar(x0-lx-5,y1-Ydot-1,x0-lx-3,y1);

    for (xp=0;xp<=Xmax;xp++)
    {
        x1=x0-Xdot2*xp;
        ht=cht[xp][yp];
        if(Fhtabs == 'Y') ht=fabs((double)ht);
        ith=-1;
        do
        {
            ith++;
        } while(((ht<fthre[ith]) && (ht<fthre[ith+1])) && (ith<8) );

        setfillstyle(SOLID_FILL,pcolor[ith]);
        bar(x1-Xdot+1,y1-Ydot+1,x1,y1);
    }
    /* end of for xp */
}
/* end of for yp */
setfillstyle(SOLID_FILL,BLACK);
bar(x0+3,y0-ly-1,x0+5,y0);
bar(x0-lx-5,y0-ly-1,x0-lx-3,y0);
}
/* end of dispimg() */

showall()
{
    int    x;
    int    y;

    initplot();
    dispimg();

```

```

    dispcmd(WHITE);
}                                     /* end of showall()          */

takesample(pecho,pht,pgood,pntotal)
unsigned int    *pecho;
float          *pht;
unsigned int    *pgood;
unsigned int    *pntotal;

{
    signed int    i,j;
    signed int    err;
    unsigned int    goodsig;
    unsigned int    ntotal;
    unsigned int    echo;
    float          echoht;

    if (Nsignal == 1 )
    {
        nsample(Nsignal,Length,Israte,Start,Trigger,
                sig,&goodsig,&ntotal);
    }
    else
    {
        err=MAXERR*2;
        for (i=0;( i<MAXN ) && ( err >MAXERR );i++)
        {
            nsample(Nsignal,Length,Israte,Start,Trigger,
                    sum,&goodsig,&ntotal);
            err=(100*(ntotal-goodsig))/ntotal;
        }
        for (j=0;j<Length;j++) sig[j]=(sum[j]+goodsig/2)/goodsig;
    }
    findlecho(sig,&echo,&echoht);
    *pecho=echo;
    *pht=echoht;
    *pgood=goodsig;
    *pntotal=ntotal;
}                                     /* end of takesample()      */

clearcursor()
{
    int          i;

    setviewport(w1l,w1t,w1r,w1b,w1clip);
    setfillstyle(SOLID_FILL,BLACK);
    bar(x0+3,  y0-ly-1,x0+5,  y0);
}

```

```

    bar(x0-lx-5,y0-ly-1,x0-lx-3,y0);
    bar(x0-lx, y0+3, x0, y0+4);
    bar(x0-lx, y0-ly-4,x0, y0-ly-3);
} /* end of clearcursor() */

dispcmd(int color)
#define CMD1 "L:load S:save T:threshold Z:size A:abs "
#define CMD2 "H:chg htmax R:reset threshold"
#define CMD3 "Space:hide command line X:chg color "
#define CMD4 "N:run again ESC:exit"
{
    int x,y;

    setviewport(w2l,w2t,w2r,w2b,w2clip);
    x=10;
    y=10;
    setcolor(color);
    gprintf(&x,&y,"%s%s",CMD1,CMD2);
    gprintf(&x,&y,"%s%s",CMD3,CMD4);
} /* end of dispcmd() */

change_color()
{
    int texth,textw;
    int i,j,k;
    int x,y;
    int ith;
    int flag2;
    char ans2,ans3;

    setviewport(w2l,w2t,w2r,w2b,w2clip);
    clearviewport();
    textw=textwidth("H");
    texth=textheight("H");
    x=10;
    y=10;
    setcolor(WHITE);
    gprintf(&x,&y," 0: 1: 2: 3: 4: 5: 6: 7:");
    gprintf(&x,&y,"");
    gprintf(&x,&y," 8: 9: A: B: C: D: E: F:");
    for (i=0;i<8;i++)
    {
        k=textw*(i+1)*7;
        setfillstyle(SOLID_FILL,i);
        bar(k+1,x+1,k+3*textw-1,x+texth-1);

        setcolor(BROWN);
        rectangle(k,x,k+3*textw,x+texth);
    }
}

```

```

    k=textw*(i+1)*7;
    setfillstyle(SOLID_FILL,i+8);
    bar(k+1,x+2*textth+5,k+3*textw-1,x+3*textth+4);

    setcolor(BROWN);
    rectangle(k,x+2*textth+4,k+3*textw,x+3*textth+4);
}

setviewport(w0l-20,w0t,w0r,w0b,w0clip);
setcolor(GREEN);
ith=0;
y=66;
x=1;
gprintf(&x,&y,"=>");
flag2=1;
while(flag2==1)
{
    do {
    } while (kbhit() ==0 );
    ans2=toupper(getch());
    switch(ans2)
    {
        case '\r':
            flag2=0;
            setcolor(BLACK);
            y=66+ith*20;
            gprintf(&x,&y,"=>");
            setviewport(w2l,w2t,w2r,w2b,w2clip);
            clearviewport();
            dispcmd(WHITE);
            break;
        case '0':
        case '1':
        case '2':
        case '3':
        case '4':
        case '5':
        case '6':
        case '7':
        case '8':
        case '9':
            pcolor[ith]=atoi(&ans2);
            k=60+ith*20;
            setfillstyle(SOLID_FILL,pcolor[ith]);
            bar(23,k,72,k+19);
            break;
        case 'A':
        case 'B':

```

```

case 'C':
case 'D':
case 'E':
case 'F':
    pcolor[ith]=ans2-55;
    k=60+ith*20;
    setfillstyle(SOLID_FILL,pcolor[ith]);
    bar(23,k,72,k+19);
    break;
case '\0':
    ans3=getch();
    switch(ans3)
    {
        case 'P':                                /* down          */
            setcolor(BLACK);
            y=66+ith*20;
            gprintf(&x,&y,"=>");
            if (ith < 8 ) ith++;
            setcolor(GREEN);
            y=66+ith*20;
            gprintf(&x,&y,"=>");
            break;
        case 'H':                                /* up              */
            setcolor(BLACK);
            y=66+ith*20;
            gprintf(&x,&y,"=>");
            if (ith > 0 ) ith--;
            setcolor(GREEN);
            y=66+ith*20;
            gprintf(&x,&y,"=>");
            break;
    }
    break;
}
}
}
}
    dispimg();
}
/* end of changecolor() */

```

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] R. Kuc, M. Schwartz, and L. V. Micsky, "Parametric estimation of the acoustic attenuation coefficient slope for soft tissue," in *IEEE Ultrasonics Symposium Proceedings*, pp. 44 – 47, 1976.
- [2] K. R. Erikson, F. J. Fry, and J. P. Jones, "Ultrasound in medicine - a review," *IEEE Transactions on Sonics and Ultrasonics*, vol. SU-21, pp. 144 – 164, July 1974.
- [3] T. E. Preuss and G. Clark, "Use of time-of-flight c-scanning for assessment of impact damage in composites," *Composites*, vol. 19, pp. 145 – 148, Mar. 1988.
- [4] D. Ensminger, *Ultrasonics-fundamentals, technology, applications*. New York: Marcel Dekker, Inc., 2nd ed., 1988.
- [5] F. Dunn, "Attenuation and speed of ultrasound in lung: Dependence upon frequency and inflation," *Journal of Acoustical Society of America*, vol. 80(4), pp. 1248 – 1250, Oct. 1986.
- [6] C. Barnes, J. A. Evans, and T. J. Lewis, "A broad-band single pulse technique for ultrasound absorption studies of aqueous solutions in the frequency range 200 kHz-1 MHz," *Ultrasonics*, vol. 24, pp. 267 – 272, Sept. 1986.
- [7] R. W. P. King and J. C. W. Harrison, "The transmission of electromagnetic waves and pulses into the earth," *Journal of Applied Physics*, vol. 39, pp. 4444 – 4452, Aug. 1968.
- [8] L. C. Chan, J. L. Peters, and D. L. Moffatt, "Improved performance of a subsurface radar target identification system through antenna design," *IEEE Transactions on Antennas and Propagation*, vol. AP-29, pp. 307 – 311, Mar. 1981.
- [9] H. J. McSkimin, "Pulse superposition method for measuring ultrasonic wave velocities in solids," *Journal of Acoustical Society of America*, vol. 33, pp. 12 – 16, Jan. 1961.
- [10] E. P. Papadakis, "Ultrasonic velocity and attenuation: Measurement methods with scientific and industrial applications," in *Physical Acoustics, Vol. XII* (W. P. Mason and R. N. Thurston, eds.), pp. 277–374, Academic, 1976.
- [11] N. P. Cedrone and D. R. Curran, "Electronic pulse method for measuring the velocity of sound in liquids and solids," *Journal of Acoustical Society of America*, vol. 26, pp. 963 – 966, Nov. 1954.

- [12] D. H. Dameron, "Determination of the acoustic velocity in tissues using an inhomogeneous media model," *IEEE Transactions on Sonics and Ultrasonics*, vol. SU-26, pp. 69 – 74, Mar. 1979.
- [13] M. V. Zummeren, D. Young, C. H. G. Baum, and R. Treleven, "Automatic determination of ultrasound velocities in planar materials," *Ultrasonics*, vol. 25, pp. 188 – 294, Sept. 1987.
- [14] J. Toulouse, "A modified version of the phase sensitive technique for measurements of absolute sound velocity in solids," in *IEEE Ultrasonics Symposium Proceedings*, pp. 407 – 410, 1987.
- [15] J. P. Weight, "A model for the propagation of short pulses of ultrasound in a solid," *Journal of Acoustical Society of America*, vol. 81(4), pp. 815 – 826, Apr. 1987.
- [16] G. C. Steyer, R. Singh, and D. R. Houser, "Alternative spectral formulations for acoustic velocity measurement," *Journal of Acoustical Society of America*, vol. 81(4), pp. 1955 – 1961, Apr. 1987.
- [17] H. J. McSkimin, "Measurement of ultrasonic wave velocities and elastic moduli for small solid specimens at high temperatures," *Journal of Acoustical Society of America*, vol. 31, pp. 287 – 295, Mar. 1959.
- [18] J. Ophir, "Estimation of the speed of ultrasound propagation in biological tissues: a beam-tracking method," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. UFFC-33, pp. 359 – 368, July 1986.
- [19] T. Kontonassios and J. Ophir, "Variance reduction of speed of sound estimation in tissues using the beam tracking method," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. UFFC-34, pp. 524 – 530, Jan. 1987.
- [20] J. Ophir and T. Lin, "A calibration-free method for measurement of sound speed in biological tissue samples," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. UFFC-35, pp. 573 – 577, Jan. 1988.
- [21] J. Ophir, Y. Yazdi, T. S. Lin, and D. P. Shattuck, "Optimization of speed-of-sound estimation from noisy ultrasonic signals," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 36, pp. 16 – 24, Jan. 1989.
- [22] R. Kuc, "Estimating acoustic attenuation from reflected ultrasound signals: Comparison of spectral-shift and spectral-difference approaches," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-32, pp. 1 – 6, Feb. 1984.
- [23] R. Kuc, "Bounds on estimating the acoustic attenuation of small tissue regions from reflected ultrasound," *Proceedings of the IEEE*, vol. 73, pp. 1159 – 1168, July 1985.

- [24] R. Kuc, "Estimating reflected ultrasound spectra from quantized signals," *IEEE Transactions on Biomedical Engineering*, vol. BME-32, pp. 105 – 112, Feb. 1985.
- [25] K. Matsuzawa, N. Inoue, and T. Hasegawa, "A new simple method of ultrasonic velocity and attenuation measurement in a high absorption liquid," *Journal of Acoustical Society of America*, vol. 81(4), pp. 947 – 951, Apr. 1987.
- [26] R. L. Roderick and R. Truell, "The measurement of ultrasonic attenuation in solids by the pulse technique and some results in steel," *Journal of Applied Physics*, vol. 23, pp. 267 – 279, Feb. 1952.
- [27] A. C. Kak and K. A. Dines, "Signal processing of broadband pulsed ultrasound: Measurement of attenuation of soft biological tissues," *IEEE Transactions on Biomedical Engineering*, pp. 321 – 344, July 1978.
- [28] Y. Hayakawa, T. Wagai, K. Yosioka, T. Inada, T. Suzuki, H. Yagami, and T. Fujii, "Measurement of ultrasound attenuation coefficient by a multifrequency echo technique- theory and basic experiments," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. UFFC-33, pp. 759 – 764, Nov. 1986.
- [29] P. He, "Acoustic attenuation estimation for soft tissue from ultrasound echo envelope peaks," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. UFFC-36, pp. 197 – 203, Jan. 1989.
- [30] G. A. McDaniel, "Ultrasonic attenuation measurements on excised breast carcinoma at frequencies from 6 to 10 MHz," in *IEEE Ultrasonics Symposium Proceedings*, pp. 234 – 236, 1977.
- [31] K. J. Parker and R. C. Waag, "Measurement of ultrasonic attenuation within regions selected from b-scan images," *IEEE Transactions on Biomedical Engineering*, vol. BME-30, pp. 431 – 437, Aug. 1983.
- [32] W. A. Verhoef, M. J. T. M. Cloostermans, and J. M. Thussen, "Diffraction and dispersion effects on the estimation of ultrasound attenuation and velocity in biological tissues," *IEEE Transactions on Biomedical Engineering*, vol. BME-32, pp. 521 – 529, July 1985.
- [33] K. J. Parker, M. S. Asztely, R. M. L. E. A. Schenk, and R. C. Waag, "Frequency dependent attenuation in normal and diseased livers," in *IEEE Ultrasonics Symposium Proceedings*, pp. 793 – 795, 1986.
- [34] E. Walach, A. Shmulewitz, Y. Itzhak, and Z. Heyman, "Local tissue attenuation images based on pulsed-echo ultrasound scans," *IEEE Transactions on Biomedical Engineering*, vol. 36, pp. 211 – 221, Feb. 1989.

- [35] N. H. Wang, "Remote sensing by acoustic video pulse techniques," Master's thesis, Michigan State University, Department of Electrical Engineering, 1988.
- [36] N. H. Wang, J. Nodar, B. Ho, and R. Zapp, "Video pulse techniques for ultrasonic sensing," in *International Symposium on Ultrasonic Imaging and Tissue Characterization*, 1988.
- [37] B. Ho, D. Ye, N. H. Wang, J. Nodar, and R. Zapp, "High range resolution ultrasonic imaging for evaluation of layered composite materials," in *Materials Research Society Conference*, 1988.
- [38] N. H. Wang, B. Ho, and R. Zapp, "Attenuation and velocity imagings of biological tissues by broadband ultrasonic signals," in *International Symposium on Ultrasonic Imaging and Tissue Characterization*, 1990.
- [39] N. H. Wang, B. Ho, and R. Zapp, "Nondestructive evaluation of material properties by broadband signal technique," in *5th Technical Conference of the American Society for Composites*, 1990.
- [40] N. H. Wang, B. Ho, and R. Zapp, "Velocity-density product and attenuation-density ratio measurements using broadband signals," *IEEE Transactions on Instrumentation and measurement*, Dec. 1991. (accepted for publication).
- [41] B. Ho, D. Ye, R. Zapp, and N. H. Wang, "Three-dimensional damage assessment in composites by ultrasonic imaging techniques," in *43rd annual conference of Reinforced Plastics and Composites*, 1988.
- [42] W. Sachse, B. Castagnede, I. Grabec, K. Y. Kim, and R. L. Weaver, "Recent developments in quantitative ultrasonic nde of composites," *Ultrasonics*, vol. 28, pp. 97 – 104, Mar. 1990.
- [43] J. A. Johnson, B. A. Barna, L. S. Beller, S. C. Taylor, and B. Walter, "A camac-based ultrasonic data-acquisition workstation," *Materials Evaluation*, pp. 934 – 938, Aug. 1987.
- [44] S. Lees, "Ultrasonic measurement of thin layers," *IEEE Transactions on Sonics and Ultrasonics*, vol. su-18, pp. 81 – 86, Apr. 1971.
- [45] I. Beretsky and G. A. Farrell, "Improvement of ultrasonic imaging and media characterization by frequency domain deconvolution, experimental study with non-biological models," *Ultrasound in Medicine*, vol. 38, pp. 1645 – 1665, 1977.
- [46] T. Sato, Y. Nakamura, O. Ikeda, and M. Hirama, "Resolution enhancement of a sector-scan image using a homomorphic transform and deconvolution," *Journal of Acoustical Society of America*, vol. 75(1), pp. 265 – 267, Jan. 1984.

- [47] J. P. Steiner, E. S. Furgason, and W. L. Weeks, "Robust deconvolution of correlation functions," in *IEEE Ultrasonics Symposium Proceedings*, pp. 1031 – 1035, 1987.
- [48] A. Yamada, "On-line deconvolution for the high resolution ultrasonic pulse-echo measurement with narrow-band transducer," in *IEEE Ultrasonics Symposium Proceedings*, pp. 1027 – 1030, 1987.
- [49] D. F. Elliott, ed., *Handbook of digital signal processing*, ch. 10. San Diego: Academic Press, Inc, 1987.
- [50] W. Mendenhall, R. L. Scheaffer, and D. D. Wackerly, *Mathematical Statistics with Applications*. Boston: Duxbury Press, 3rd ed., 1986.
- [51] E. E. Hundt and E. A. Trautenberg, "Digital processing of ultrasonic data by deconvolution," *IEEE Transactions on Sonics and Ultrasonics*, vol. su-27, pp. 249 – 252, Sept. 1980.
- [52] T. Yokota and Y. Sato, "Super-resolution ultrasonic imaging by using adaptive focusing," *Journal of Acoustical Society of America*, vol. 77(2), pp. 567 – 572, Feb. 1985.
- [53] D. P. Shattuck, M. D. Weinshenker, S. W. Smith, and O. T. V. Ramm, "Explososcan: a parallel processing technique for high speed ultrasound imaging with linear phased arrays," *Journal of Acoustical Society of America*, vol. 75(4), pp. 1273 – 1282, Apr. 1984.
- [54] S. W. Smith, J. H. G. Pavy, and O. T. V. Ramm, "High-speed ultrasound volumetric imaging system - part I: Transducer design and beam steering," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 38, pp. 100 – 108, Mar. 1991.
- [55] O. T. von Ramm, S. W. Smith, and J. H. G. Pavy, "High-speed ultrasound volumetric imaging system - part II: Parallel processing and imaging display," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 38, pp. 109 – 115, Mar. 1991.
- [56] J. F. Greenleaf and R. C. Bahn, "Clinical imaging with transmissive ultrasonic computerized tomography," *IEEE Transactions on Biomedical Engineering*, vol. 28, pp. 177 – 185, 1981.
- [57] J. F. Greenleaf, "Computerized tomography with ultrasound," *Proceedings of the IEEE*, vol. 71, pp. 330 – 337, 1983.
- [58] H. Ermert and G. Rohrlein, "Ultrasound reflection-mode computerized tomography for in-vivo imaging of small organs," in *IEEE Ultrasonics Symposium Proceedings*, pp. 825 – 828, 1986.

- [59] W. D. Richard, "A new time-gain correction method for standard B-mode ultrasound imaging," *IEEE Transactions on Medical Imaging*, vol. 8, pp. 283 – 285, Sept. 1989.
- [60] R. Momenan, M. H. Loew, R. F. Wagner, M. F. Insana, and B. S. Garra, "Application of pattern recognition techniques in ultrasound tissue characterization," in *IEEE Engineering in Medicine & Biology Society Conference*, pp. 411–412, 1989.
- [61] D. Brzakovic, X. M. Luo, and P. Brzakovic, "An approach to automated detection of tumors in mammograms," *IEEE Transactions on Medical Imaging*, vol. 9, pp. 233 – 241, Sept. 1990.
- [62] S. M. Lai, X. Li, and W. F. Bischof, "On techniques for detecting circumscribed masses in mammograms," *IEEE Transactions on Medical Imaging*, vol. 8, pp. 377 – 386, Dec. 1989.
- [63] M. Hashimoto, P. V. Sankar, and J. Sklansky, "Detecting the edges of lung tumors by classification techniques," in *IEEE Ultrasonics Symposium Proceedings*, pp. 276 – 279, 1982.
- [64] M. Bomans, K. Hohne, U. Tiede, and M. Riemer, "3-D segmentation of MR images of the head for 3-D display," *IEEE Transactions on Medical Imaging*, vol. 9, pp. 177 – 183, June 1990.
- [65] S. P. Raya, "Low-level segmentation of 3-d magnetic resonance brain images - a rule-based system," *IEEE Transactions on Medical Imaging*, vol. 9, pp. 327 – 337, Sept. 1990.
- [66] S. S. Trivedi, G. T. Herman, and J. K. Udupa, "Segmentation into three classes using gradients," *IEEE Transactions on Medical Imaging*, vol. 5, pp. 116 – 119, June 1986.
- [67] J. W. K. Jr., C. L. Vaughan, T. D. F. Jr., and L. T. Andrews, "Segmentation of echocardiographic images using mathematical morphology," *IEEE Transactions on Biomedical Engineering*, vol. 35, pp. 925 – 934, Nov. 1988.
- [68] D. J. Michael and A. C. Nelson, "Handx: a model-based system for automatic segmentation of bones from digital hand radiographs," *IEEE Transactions on Medical Imaging*, vol. 8, pp. 64 – 69, Mar. 1989.
- [69] A. Rosenfeld, "Computer vision: a source of models for biological visual processes," *IEEE Transactions on Biomedical Engineering*, vol. BME-36, pp. 93 – 96, Jan. 1989.
- [70] B. A. Auld, *Acoustic Fields and Waves in Solids, Vol. I*. New York: John Wiley & Sons, Inc., 1973.
- [71] V. M. Ristic, *Principles of Acoustic Devices*. New York: John Wiley & Sons, Inc., 1983.

- [72] S. Ramo, J. R. Whinnery, and T. V. Duzer, *Fields and waves in communication electronics*. New York: John Wiley & Sons, Inc., 2nd ed., 1984.
- [73] N. R. Draper and H. Smith, *Applied Regression Analysis*. New York: John Wiley & Sons, Inc., 2nd ed., 1981.
- [74] L. W. Johnson and R. D. Riess, *Numerical analysis*. Reading, Massachusetts: Addison-Wesley Publishing Company, 2nd ed., 1982.
- [75] N. H. Wang, J. T. Sheu, and B. Ho, "Tissue characterization by hierarchical clustering techniques," in *International Symposium on Ultrasonic Imaging and Tissue Characterization*, 1991.
- [76] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. New Jersey: Prentice-Hall, Inc., 1988.
- [77] J. C. Birnholz, "An approach to specific tumor diagnosis by ultrasound imaging," in *IEEE Ultrasonics Symposium Proceedings*, 1972.

MICHIGAN STATE UNIV. LIBRARIES



31293009063516