(13)

The Pennsylvania State University
APPLIED RESEARCH LABORATORY
P.O. Box 30
State College, PA 16804

FEASIBILITY FOR ULTRASONIC CHARACTERIZATION OF THE SURFACE ROUGHNESS OF ATHEROSCLEROTIC PLAQUE

by

D. D. Hughes

Technical Report No. TR 94-04
February 1994

Supported by:
National Institutes of Health

L.R. Hettche, Director
Applied Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	February 1994	

4. TITLE AND SUBTITLE

Feasibility for Ultrasonic Characterization of the Surface Roughness of Atherosclerotic Plaque

5. FUNDING NUMBERS

N00039-88-C-0051

6. AUTHOR(S)

D. D. Hughes

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Applied Research Laboratory
The Pennsylvania State University
P.O. Box 30
State College, PA 16804

8. PERFORMING ORGANIZATION REPORT NUMBER

TR#94-04

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Institutes of Health

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This study documents the investigation of ultrasound as a method for diagnosis of atherosclerotic lesions. The study discusses the theoretical background for the use of backscattered radiation on random rough surfaces. The Kirchhoff approximation is used to model the return intensity. Sample surfaces made of sanded Plexiglas, glass beads and aluminum casting are used to compare experimental results with theory prediction. Several arterial tissue samples are used to demonstrate the technique.

14. SUBJECT TERMS

Ultrasound, Diagnosis, Atherosclerotic lesions, Backscattered radiation, Surface roughness

15. NUMBER OF PAGES

142

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

ABSTRACT

This study documents the investigation of ultrasound as a method for diagnosis of atherosclerotic lesions. The study discusses the theoretical background for the use of backscattered radiation on random rough surfaces. The Kirchhoff approximation is used to model the return intensity. Sample surfaces made of sanded Plexiglas, glass beads and aluminum casting are used to compare experimental results with theory prediction. Several arterial tissue samples are used to demonstrate the technique.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Atherosclerosis is a condition of surface degradation of the lining of the arteries. This condition can lead to heart attack or stroke. Heart attacks alone cause 500,000 deaths a year in the United States. The detection of atherosclerosis early in its development can improve the prognosis of the individual with the condition. The use of ultrasound is one method to detect atherosclerosis. The thesis consists of six parts. The first part describes the disease and current ultrasound modalities used for the diagnosis of the disease. The second part discusses the theoretical basis of scattering of waves from rough surfaces and how it can be applied to studying atherosclerosis. The third part discusses the characteristics of phantoms and tissues that were experimentally studied. The fourth section discusses the equipment and procedures used. The fifth section presents the results. The sixth section contains the conclusions. The appendices contain supporting data and the software used for collecting and analyzing data.

Atherosclerosis

To discuss the disease some background is necessary of the structure of an artery. The artery is composed of three layers of tissue (Gray, 1974, p. 1126). The innermost layer, the *tunica intima*, itself also contains three parts. The

innermost part is composed of smooth epithelial cells oriented longitudinally to the vessel. These epithelial cells serve as a barrier for blood components and a boundary surface that has little drag to impede the blood flow. Also part of the intima is a subendothelial layer of connective tissue with branched cells in the interspaces of the tissue, a layer which is more developed in the larger vessels. The chief thickness of the intima is due to a third part which is composed of elastic longitudinal fibers with small elongated perforations. The next layer is the *tunica media.* The media is composed of muscle cells that are oriented circumferentially to the vessel. These muscles control the diameter of the artery depending on the required needs of the organs served by the artery. The outermost layer is the *tunica adventitia*, composed of connective tissue that maintains the integrity of the artery and its maximum diameter. This layer is largely absent in the larger arteries.

The exact causes of the disease are not completely defined, although some contributing factors are known. Chazov and Smirnov documented five problems that researchers have directed their efforts at solving (1982, p. viii):

1. The mechanism regulating cell and chemical homeostasis.
2. Molecular mechanisms of endothelial injury and its repair.
3. Mechanisms of intima hyperplasia.
4. Lipidogenic and thrombogenic mechanisms of plaque formation.
5. Origin of foam cells and pathways of extracellular matrix formation in the plaque.

These are all basic areas for research. The development of the disease is assumed to come from two etiologies (Likar and Robinson, 1985, p. 12). The first is insudation, the passage and accumulation of plasma components into the intima. The second etiology is encrustation, the accumulation of thrombi

or blood clots at points of arterial weakness or damage which can occur from a number of causes. Atherosclerosis may result in plaques, sometimes occluding the vessel, containing a matrix loaded with lipids (Likar and Robinson, 1985, p. 10). In general the development of the atherosclerotic condition begins with a thickening of the intima. This is accompanied with changes in the media, either local or diffuse depending on external factors such as blood pressure, thickening rate, surrounding tissue and type of metabolic disorder. This increased growth is not accompanied by growth of the *vasa vasorum*, nutrient vessels supplying the larger blood vessels. This loss of nutrient results in regressive metamorphosis (Likar and Robinson, 1985, p. 11). Attempts to repair the tissue lead to clot formation and further thickening with the clots having a fibrous nature. Ultimately, the vessel becomes partially or totally occluded, also accompanied by calcification which is characterized by granular or plate-like shells in the atheroma. These developmental stages are sometimes used as references for characterization of diseased tissue. Barzilai et al. used four stages of development: normal, fibrous, fibrofatty, and calcified (1987, p. 460). Picano et al. used five stages: normal, fatty, fibrofatty, fibrotic, and calcified (1985, p. 572).

Another more recent appraisal of the development of atherosclerosis is found in a 1991 report published by the U. S. Department of Health and Human Services (p. 9). The deposition of lipid begins in the arteries of children as fatty streaks. The extent of the fatty streaks increases in adolescence and early adulthood. Fibrous plaques (larger lipid deposits with a covering of smooth muscle and connective tissue) also begin to appear and increase during the second and third decades. Some of the plaques develop other changes

such as vascularization, hemorrhage, and ulceration. These predispose the vessels to thrombosis during middle age and later.

Imaging of Atherosclerosis

Diagnosis of atherosclerosis is dependent on the ability to image the lesions clinically. There are both noninvasive and invasive intravascular techniques for imaging atherosclerosis. And for both techniques there are several modalities, each having some advantages and disadvantages.

Noninvasive techniques include contrast angiography and B-mode ultrasound. Angiography involves the injection of a radiopaque dye into the vascular system and using a fluoroscope or x-ray to visualize the artery. This can be real-time with fluoroscopy, or a single image or set of images with radiography. The technique can detect stenosis, or narrowing of the artery, which can be measured and compared with adjacent areas of the vessel. The technique has some disadvantages: it can not distinguish the type of atherosclerotic lesion, and the actual measurement may not be the maximum stenosis due to orientation of the image plane, as most lesions are eccentric in cross-section. This can however, be partially corrected by orienting the patient when using fluoroscopy. There is also the clinical risk associated with injection of contrast media (Wells, 1982, pp. 46-48).

Imaging of atherosclerotic plaque is possible with noninvasive ultrasound. B-mode ultrasound imaging presents a display with one axis representing azimuth and the other axis representing range (Institute of Electrical and Electronic Engineers, Inc., 1982, p. 10). The transducer is placed against the

body with an impedance matching gel to insure good transmission between the probe and a vessel. This technique also can image aneurysm and thrombi, which may not always be possible with x-ray (Wells, 1982, p. 178). There are disadvantages associated with this technique also. The vessel is often not easily imaged because of intervening acoustically opaque tissue, such as bone or the air in lungs. Additionally, attenuation becomes significant for imaging vessels farther away from the surface, especially for higher frequency probes that allow better resolution images. The images of atherosclerotic lesions by B-mode scan have had limited success thus far in characterizing biological composition and surface of a lesion.

Intravascular techniques are a more recent development. As well as usage in diagnosis they are often used in conjunction with atherectomy, balloon angioplasty and laser angioplasty. There are two intravascular techniques currently being developed: fiberoptic angioscopy and intravascular ultrasound (Coy et al., 1991, pp. 1811-1812). Both of these techniques are delivered by catheter through the arteries.

Fiberoptic angioscopy provides visual evidence of thrombus in acute ischemia syndromes, areas of blood deficiency. It also allows the inspection of the vessel for grafting suitability and placement. Additionally, the surface of the intima can be inspected and any intraluminal flaps can be detected. Unfortunately, these imaging capabilities can only be done while flushing the area of any blood with clear saline. Another disadvantage is the inability to image beyond the intima.

Intravascular ultrasound uses two kinds of transducer arrangements: phased array and rotating mirror. Each system views the structure of an

artery in cross-section including intima, media, and adventitia, as well as some external tissue.

The images of these layers bear a relationship to the structure of the artery. The image of a normal artery shows a bright return from the intima and the internal elastic lamina. The next layer is the non-echogenic media, which appears as a dark region. A third layer, bright again, corresponds to the adventitia (Yock et al., 1991 p 39B). This approach appears to produce best images in muscular arteries, which are generally the ones most often found with atherosclerosis. The thicknesses of these regions have shown good agreement with histologic measurement with the exception of the single intima cell layer because it is smaller than the resolution of most intravascular ultrasonic imaging catheters.

Because of the close proximity to the lesions very high frequencies (20 - 30 megahertz) can be used with the attendant higher resolution without loss due to attenuation. This also overcomes the disadvantage of noninvasive ultrasound of intervening air and bone structures. By moving the catheter along the artery, three-dimensional structure can be seen. This allows the presence and structure of atherosclerosis to be defined both beneath the surface and along the vessel. The limitation of the diameter of the vessel is present, but catheters as small as 1.16 millimeter are used (Coy et al., 1991, p. 1812). This limitation restricts the use to lumen smaller than this diameter and means that the normal diameter of the artery will have to be greater depending on the degree of stenosis. However, because of the high frequencies (20 to 30 megahertz), imaging can take place with the catheter nearly touching the intima. The anticipated use of intravascular ultrasound is for diagnosis as well as positioning of angioplasty devices, and assessment of restenosis after

their use. While there are many advantages there are also some disadvantages. The scanning is perpendicular to the axis of the vessel which prevents a view of what possible obstructions might be ahead, possibly leading to some mechanical trauma to the vessel.

Purpose of Thesis

In a recent U. S. Department of Health and Human Services report on atherosclerosis a recommendation was made to "support a critical appraisal of newer techniques" the first of which is intravascular ultrasound (1991, p. 127). Among the goals set forth was "to image the structure and composition of the atheroma." It is the purpose of this thesis to investigate the possibility of using ultrasound backscatter for the characterization of atherosclerosis. The assumption is that, as with radar mapping of the ground from aircraft, oblique incidence waves can characterize the surface without being overwhelmed by specular return.

Externally this could be done with basically conventional imaging approaches, the only difference being an off axis transducer to transmit the signal. Internal usage would require the use of a catheter-mounted array or similar device. It is proposed that a forward scanning transducer could provide this capability intravascularly.

The rationale behind this thesis is that varying degrees of roughness will scatter incoming energy over differing angular spread and at different levels. A similar principle has been used in radar, sonar, and other mapping methods. Specular scattering or reflection is the most direct return of energy from a

surface or object. If an acoustic beam is directed to a smooth surface perpendicular to it, the energy will return along the same path. If the surface is not perpendicular to the beam the energy will return along an angle that is equal to the incident angle. A perfectly smooth surface will scatter or reflect only in the specular direction. As roughness increases scattering begins to spread angularly. It is this energy that is the basis for sonar and radar mapping. In medical ultrasound the specular return is used for characterizing boundary of tissues of different properties. The signal level returned from an interface between two inhomogeneous materials is so high that it is sometimes difficult to retrieve any characteristic of the boundary. With an off-axis method in which the effect of the main clutter is minimal, differences may become apparent.

Chapter 2

BACKSCATTER FROM ROUGH SURFACES

An analysis of scattering of various forms of waves from surfaces is important in many fields. The most common field is radar where the backscatter of radar energy allows the mapping of surface features for navigation and mapping, detection of moisture for weather warning and prediction, and the detection of aircraft for military as well as commercial purpose. Scattering is also used as the basis of one form of X-ray imaging in medicine (Shung et al., 1992, p. 62). Acoustic energy is used in the oceans for detection of various submerged objects from enemy submarines to schools of fish. Acoustic energy is also used in seismic exploration for oil and other studies of the morphology of the earth. A discussion of the theoretical basis for scattered energy from surfaces is necessary to understand whether atherosclerotic plaque could be imaged. Since the plaque surface roughness characteristic is virtually unknown, it has been assumed to be a Gaussian distribution.

Theoretical Background

Any discussion of scattering from rough surfaces must mention that research began with Lord Rayleigh's investigation of normal incident scattering from a sinusoidal surface in 1877 (Volume 2, 1945, p.89-96). From that beginning to more recent developments the direction taken has involved a move

from close formed solutions to those that satisfy random roughness conditions. Ogilvy's recently published book has the clearest presentation of the necessary theoretical development and will be used as the primary reference to what follows (1991).

One of the outcomes of Lord Rayleigh's work was the generally used term, Rayleigh criterion, as a description of a surface's smoothness or roughness. The geometry of plane wave incidence is illustrated in Figure 2.1. The incident energy arrives at an angle, θ_1, from the normal to the average height plane. The incident energy that scatters in the same direction, θ_2, from two different points will have some change in phase, $\Delta\phi$.

$$\Delta\phi = k[(h_1 - h_2)(\cos\theta_1 + \cos\theta_2) + (x_2 - x_1)(\sin\theta_1 - \sin\theta_2)] \qquad (2.1)$$

The term, k, is the wave number ($2\pi/\lambda$; λ, wavelength) of the incident wave. If $\Delta\phi$ is very small compared to π, the scattered waves will be nearly in phase and will constructively interfere. If $\Delta\phi$ is approximately equal to π, the scattered waves will be out of phase and will destructively interfere. The Rayleigh criterion limits $\Delta\phi$ to values less than $\pi/2$ for smooth surfaces. A value larger than $\pi/2$ indicates a rough surface (Ogilvy, 1991, p. 3). Replacing the difference in h with the average root mean square (RMS) surface deviation, σ, the criterion becomes:

$$R_a = k\sigma\cos\theta_1 < \pi/4. \qquad (2.2)$$

The symbol, R_a, is called the Rayleigh parameter. In the general literature for material roughness this same symbol is used for the average roughness of a surface, usually a machined surface (Bennett and Mattsson, 1989, p. 39). In this thesis average roughness will be symbolized by R_A.

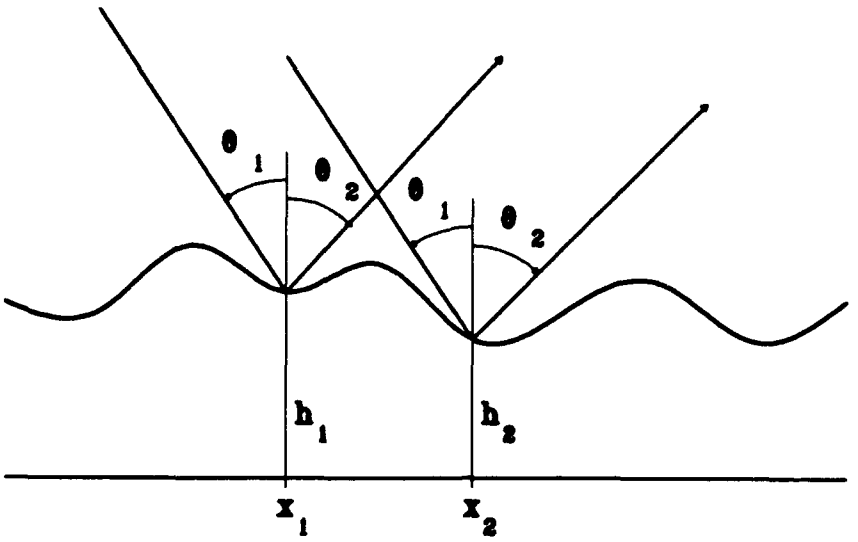

Figure 2.1. Rough scattering from two points.

In the discussion of roughness an important concept is that roughness is not an intrinsic characteristic of the surface. A surface can be considered smooth or rough depending on the wavelength of the propagating energy in the fluid medium. This can be seen in the mutual relationships of the three variables in equation 2.2. The combined value of $k\sigma \cos\theta_1$ determines whether a surface is considered rough. In cases where $k\sigma \cos\theta_1$ is nearly $\pi/4$ the value of θ may be able to shift the Rayleigh parameter from a rough designation at normal incidence to a smooth designation at nearly grazing incidence (Ogilvy, 1991, p. 4).

Another area of concern is the phase relationship of scattered wave fronts from an extended surface. From equation 2.1 for a smooth surface ($h_1 = h_2$),

the phase varies over different values of $(x_2 - x_1)$. For the case of a smooth surface, the result is the familiar beam pattern that clusters most of the energy in the specular direction, where $\theta_1 = \theta_2$. Additionally, all the energy is in-phase, constructively interfering. As a surface becomes less smooth this relationship becomes less clearly aligned in the specular direction and the beam pattern spreads in angle and becomes less powerful in the specular direction. This reduction is approximated by $e^{-g/2}$ where:

$$g = 4k^2\sigma^2 \cos^2 \theta_1 = 4R_a^2 \tag{2.3}$$

Additionally, as noted in equation 2.1, the scattering in non-specular directions has some phase relationship that is determined by the variations across the surface. Due to the random nature of the surface fluctuations the phase relationship across a surface will be distributed across all possible values. The total pressure level from this scattering will then have an integrated amplitude that sums to zero. However, at any angle there will be some energy scattered which has a phase that is out of phase with the coherent field of the specular scattering. This energy is termed the diffuse or incoherent field (Ogilvy, 1991, pp. 4-6).

Rough surfaces by their very nature can not be represented by any closed-form description. This requires that the surfaces be described by statistical properties. To be able to use statistical theory in the analysis of scattering from rough surfaces several conditions must be satisfied.

It is often assumed the surface is isotropic. This condition says that the surface characteristics are the same in any direction chosen. While this condition is not essential, it does allow some simplification of the mathematics. A necessary condition for the use of statistical theory is that the surface

exhibit stationarity. This condition is that the statistics relating to one area are identical to any other area of the same surface. The probability that a surface has an expected value for height at one point does not change as the point is changed. The final condition for use of statistical theory is ergodicity. This is the condition that any stationary statistical surface will retain the same properties regardless of looking at a single sample or an ensemble of samples (Bendat and Piersol, 1971, pp. 10-12).

There are two statistical values that can define a surface. The first is the variation of heights along a surface or the surface height distribution. The second value is the spread of heights along the surface or the surface correlation function (Ogilvy, 1991, pp. 9-10).

Suppose that the surface height is $h(\mathbf{r})$, where h is the height above or below a reference surface, and $\mathbf{r}$ is the position vector on the surface. For this random surface there is a statistical height distribution, $p(h)$, where $p(h)dh$ is the probability that a point on the the surface has a height between h and $h + dh$ away from the mean surface. The mean surface is usually defined as:

$$\langle h \rangle_s = \int_{-\infty}^{\infty} hp(h)dh = 0 \tag{2.4}$$

where $\langle h \rangle_s$ represents the expected value or the average across a surface. The root mean square height is the standard deviation and is defined as

$$\sigma = \sqrt{\langle h^2 \rangle_s}. \tag{2.5}$$

Surface roughness can also be calculated by the arithmetic mean or center line average (cla) which is defined as

$$R_{cla} = \int_{-\infty}^{\infty} |h|p(h)dh. \tag{2.6}$$

Considerable argument about the aptness of a Gaussian assumption has been made (Ogilvy, 1991, pp. 11-12). In general, the conclusion is that it is a good assumption until proven otherwise. For a Gaussian distribution the formula is:

$$p(h) = \frac{1}{\sigma\sqrt{2\pi}} exp(-\frac{h^2}{2\sigma^2})$$

For a given Gaussian distribution $p(h)$, the center line average has the following relationship to the standard deviation (Ogilvy, 1991, p. 10):

$$R_{cla} = \sigma\sqrt{2/\pi}. \tag{2.7}$$

Measurements of surface roughness can be done by several methods. The usual method is average roughness, R_A, and is done with a profilometry device. This device moves a stylus across the surface and measures the deflections as it moves. An average surface roughness is calculated and the distance, h_i, above or below this surface is summed and then divided by the length L (Talysurf 10 Operator's Manual, 1977, p. 3).

$$R_A = \frac{h_1 + h_2 + h_3 \ldots h_n}{L} = \frac{1}{L}\int_0^L |h| dL \tag{2.8}$$

The definition for R_A is identical to the center line average. The surface roughness is also described as the root mean square roughness σ. This is defined as the square root of the mean value of squared heights from the mean surface level (J. M. Bennett and L. Mattsson, 1989, p. 38).

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N} h_i^2} \tag{2.9}$$

The values for R_A and σ are similar if there are no large deviations from the mean surface level. If there are many large deviations they will dominate the

sum and σ will be larger than R_A (J. M. Bennett and L. Mattsson, 1989, p. 39). While Ogilvy says there is a scale factor of $\sqrt{2/\pi}$, the values calculated or measured for R_A are assumed equivalent to σ in this thesis.

The second value to define the rough surface is correlation length. The RMS height does not distinguish between a rapidly changing surface and a slowly varying one. To find this length scale factor a correlation function must be used (Ogilvy, 1991, p. 13). The autocorrelation function is defined as:

$$D(\mathbf{R}) = \frac{\langle h(\mathbf{r})h(\mathbf{r}+\mathbf{R})\rangle_s}{\sigma^2} \tag{2.10}$$

This is the normalized version of the autocovariance function which lacks the σ^2 term. The autocovariance function in summation notation is:

$$D(l) = \frac{1}{N}\sum_{i=1}^{N-l} h_i h_{i+l}, l = 0, 1, 2, \ldots, N-1 \tag{2.11}$$

The values of h_i are measured from the mean surface (Bennett and Mattsson, 1989, p. 46). If a surface is Gaussian the correlation function is given by

$$D(R) = exp(-\frac{R^2}{\lambda_o^2}). \tag{2.12}$$

Here the factor λ_o is the correlation length. The correlation length is defined as the distance along the correlation function where the value drops to e^{-1} times the value at zero. Notice that the value R in equation 2.12 is no longer a vector. This is because the surface is assumed to be isotropic. The direction of correlation measurement will yield the same result for isotropic surfaces regardless of direction chosen.

An alternative description of the surface is to look at the power spectrum. This is the Fourier transform of the un-normalized correlation function.

$$P(\mathbf{k}) = \frac{\sigma^2}{(2\pi)^2}\int_{-\infty}^{\infty} D(\mathbf{R})e^{i(\mathbf{k})\cdot(\mathbf{R})}d(\mathbf{R}) \tag{2.13}$$

For a Gaussian surface with an anisotropic surface the power spectrum has the following form

$$P(k_1, k_2) = \frac{\sigma^2 \lambda_1 \lambda_2}{4\pi} exp(-\frac{k_1^2 \lambda_1^2}{4}) exp(-\frac{k_2^2 \lambda_2^2}{4}). \tag{2.14}$$

The spectrum is also Gaussian with a mean of zero and a standard deviation of $\sqrt{2}/\lambda_j$ where j takes the values $1, 2$ for the x and y directions (Ogilvy, 1991, pp. 17-18). If the surface is isotropic the formula simplifies to:

$$P(k) = \frac{\sigma^2 \lambda_o^2}{4\pi} exp(-\frac{k^2 \lambda_o^2}{4}). \tag{2.15}$$

There are some additional functions that are used in the description of surface roughness. While they are not used in this thesis they are mentioned for completeness. One such function is the surface structure function. This function is the mean square of the difference in height. This function has the advantage of being independent of the reference height chosen. Unfortunately it does not figure directly in the theory of wave scattering (Ogilvy, 1991, p. 17). Another function is the characteristic function which is the Fourier transform of the probability density function.

$$\chi(s) = \int_{-\infty}^{\infty} p(h) e^{ish} dh \tag{2.16}$$

This has the advantage of including the phase modulation of the wave at a rough surface. In addition to these two functions there are higher order surface functions. The first is the two point height probability distribution. If two sets of random variables are independent their joint probability becomes the product of their single point probability functions. There are also higher order surface correlations. Two such functions are skewness and kurtosis. Skewness is a third moment measure of how the random surface has more

peaks or valleys. A positive skewness indicates that the average surface level has been calculated higher than it should be. Kurtosis is a fourth order moment description of the surface. A kurtosis value of 3 indicates a Gaussian distribution of height. Less than three indicates a surface with fewer extremes than a Gaussian surface. A kurtosis greater than 3 indicates a surface with more small height variations than a Gaussian surface (Bennett and Mattsson, 1989, p. 43). Finally there is the slope distribution function. This measures the slope of the surface rather than the height(Bennett and Mattsson, 1989, p. 47).

There are two approximations used in the theory of scattering from rough surfaces. They are perturbation theory and tangent plane or Kirchhoff theory. In general, they cover different conditions but do have some overlap. For both theories the acoustic field is assumed to be composed of the incident and scattered fields.

$$\psi(\mathbf{r}) = \psi^{inc}(\mathbf{r}) + \psi^{sc}(\mathbf{r}) \tag{2.17}$$

The function ψ is usually the velocity potential for acoustic waves.

Perturbation theory is used for slightly rough surfaces. There are restrictions in its use:

$$k|h(x,y)| \ll 1 \tag{2.18}$$

$$|\nabla h(x,y)| \ll 1 \tag{2.19}$$

These restrictions arise from the use of Taylor series expansion on the mean scattering surface. Equation 2.18 indicates that the surface height must be much less than the wavelength. Equation 2.19 indicates that the slopes are very small, indicating a gently sloping undulation where phase relationships are generally preserved (Ogilvy, 1991, pp. 38-39).

The Kirchhoff theory makes an approximation to the field on the surface of the scatterer. Each point on the surface is assumed to be part of an infinite plane. As Ogilvy notes the theory is exact for an infinite smooth plane, but becomes an approximation for scatterers that are finite, rough or non-planar (1991, p.73). The choice of theory becomes one of selecting a theoretical base that is at best an approximation but at least is valid in the regions of surface roughness relative to the wavelength of interest.

Development for Application

The Kirchhoff theory for acoustic waves is developed for the application of modelling surfaces of atherosclerotic plaque. Using a Helmholtz integral scattering formula to compute the scattered field results in the following formula starting from equation 2.17.

$$\psi(\mathbf{r}) = \psi^{inc}(\mathbf{r}) + \int_{\mathbf{S_o}} [\psi^{sc}(\mathbf{r_o}) \frac{\partial G(\mathbf{r}, \mathbf{r_o})}{\partial \mathbf{n_o}} - G(\mathbf{r}, \mathbf{r_o}) \frac{\partial \psi^{sc}(\mathbf{r_o})}{\partial \mathbf{n_o}}] d\mathbf{S_o} \qquad (2.20)$$

In equation 2.20 $\mathbf{S_o}$ is the surface of the scatterer, $\mathbf{r}$ is in a closed volume containing no sources from $\psi^{inc}(\mathbf{r})$ or $\psi^{sc}(\mathbf{r})$ or outside a closed volume that encloses all the sources. The unit surface normal, $\mathbf{n_o}$, points towards the source. The function, $G(\mathbf{r}, \mathbf{r_o})$, is the acoustic Green's function of the field at $\mathbf{r}$ for a source at $\mathbf{r_o}$. The Green's function for full space scattering from a finite surface is:

$$G(\mathbf{r}, \mathbf{r_o}) = \frac{exp(ik|\mathbf{r} - \mathbf{r_o}|)}{4\pi|\mathbf{r} - \mathbf{r_o}|} \qquad (2.21)$$

Here, $\mathbf{r_o}$, is on the scatterer surface and $\mathbf{r}$ is some distance from the surface. When the surface $\mathbf{S_o}$ is closed the scattered field, $\psi^{sc}(\mathbf{r_o})$, in equation 2.20

is interchangeable with total field, $\psi(\mathbf{r}_o)$. This is due to no sources of ψ^{inc} within $\mathbf{S}_o$.

$$\int_{\mathbf{S}_o} [\psi^{inc}(\mathbf{r}_o)\frac{\partial G(\mathbf{r},\mathbf{r}_o)}{\partial \mathbf{n}_o} - G(\mathbf{r},\mathbf{r}_o)\frac{\partial \psi^{inc}(\mathbf{r}_o)}{\partial \mathbf{n}_o}]d\mathbf{S}_o = 0 \qquad (2.22)$$

The choice of surface depends on the field to be used in the integrand of equation 2.20. If the scattered field is used the rough surface must connect with a surface at infinity. The surface at infinity has a scattered field of zero. If the total field is used in equation 2.20, the surface is closed with a surface just below the rough surface with a total field of zero on this closing portion. The first assumption ignores the edges of finite surfaces. The second surface integral will include the effect of diffraction from the edges. Ogilvy citing Beckmann and Spizzichino, using the second method (total field) gives the scattered field as (1991, p.77):

$$\psi^{sc}(\mathbf{r}) = \psi(\mathbf{r}) - \psi^{inc}(\mathbf{r})$$

$$= \int_{\mathbf{S}_o} [\psi(\mathbf{r}_o)\frac{\partial G(\mathbf{r},\mathbf{r}_o)}{\partial \mathbf{n}_o} - G(\mathbf{r},\mathbf{r}_o)\frac{\partial \psi(\mathbf{r}_o)}{\partial \mathbf{n}_o}]d\mathbf{S}_o. \qquad (2.23)$$

Using the reflection coefficient for plane waves, R_o, the total field on the surface is (the Kirchhoff approximation):

$$\psi(\mathbf{r}_o) = [1 + R_o(\mathbf{r}_o)]\psi^{inc}(\mathbf{r}_o) \qquad (2.24)$$

Assuming a plane monochromatic wave of the form

$$\psi^{inc}(\mathbf{r}) = e^{i\mathbf{k}_{inc}\cdot\mathbf{r}} \qquad (2.25)$$

where $\mathbf{k}_{inc}$ is the incident wavevector. This sets the normal derivative as

$$\frac{\partial \psi(\mathbf{r}_o)}{\partial \mathbf{n}_o} = i[1 - R_o(\mathbf{r}_o)]\mathbf{k}_{inc}\cdot\mathbf{r}\psi^{inc}(\mathbf{r}). \qquad (2.26)$$

The next step is to make the far field approximation, $r \gg r_o$. Additional requirements are often placed on the problem for incident beams: $r \gg d^2/\lambda_o$ and $d^2 \gg \lambda_o^2$ where d is the diameter of the illuminated surface. This latter restriction is necessary also for the stochastic problem. The far field approximation results in the scattered field:

$$\psi^{sc}(\mathbf{r}) = \frac{ie^{ikr}}{4\pi r}\int_{S_o}[(R_o\mathbf{k}^- - \mathbf{k}^+)\cdot\mathbf{n}_o]e^{i\mathbf{k}^-\cdot\mathbf{r}_o}dS_o \tag{2.27}$$

where the wavevectors are defined as:

$$\mathbf{k}^- = \mathbf{k}_{\mathbf{inc}} - \mathbf{k}_{\mathbf{sc}}$$

$$\mathbf{k}^+ = \mathbf{k}_{\mathbf{inc}} + \mathbf{k}_{\mathbf{sc}} \tag{2.28}$$

The integration is over the surface, $\mathbf{S_o}$, which undulates. Converting the integration over the mean plane surface requires that the element area become:

$$\mathbf{n}_o dS_o \sim (-\hat{\mathbf{i}}\frac{\partial h}{\partial x_o} - \hat{\mathbf{j}}\frac{\partial h}{\partial y_o} + \hat{\mathbf{k}})dS_M \tag{2.29}$$

since the unit normal is

$$\mathbf{n}_o = \frac{-\hat{\mathbf{i}}\partial h/\partial x_o - \hat{\mathbf{j}}\partial h/\partial y_o + \hat{\mathbf{k}}}{\sqrt{[1+(\partial h/\partial x_o)^2 + (\partial h/\partial y_o)^2]}} \tag{2.30}$$

This change also requires the additional assumption that no point on the surface has infinite gradient (Ogilvy, 1991, p.79).

For the standard geometry shown in Figure 2.2, where θ_1 is the angle from the z axis to the incident wave vector which projects onto the x axis, θ_2 is the angle from the z axis to the scattered wave vector, and θ_3 is the angle from the x axis to the projection of the scattered wave vector onto the xy plane, the incident and scattered wave vectors become

$$\mathbf{k}_{\mathbf{inc}} = k(\hat{\mathbf{i}}\sin\theta_1 - \hat{\mathbf{k}}\cos\theta_1)$$

$$\mathbf{k_{sc}} = k(\hat{\mathbf{i}}\sin\theta_2\cos\theta_3 + \hat{\mathbf{j}}\sin\theta_2\sin\theta_3 + \hat{\mathbf{k}}\cos\theta_2) \tag{2.31}$$

The scattered field becomes

$$\psi^{sc}(\mathbf{r}) = \frac{ie^{ikr}}{4\pi r}\int_{S_M}(a\partial h/\partial x_o + b\partial h/\partial y_o - c)$$

$$\times exp[ik(Ax_o + By_o + Ch(x_o, y_o)]dx_o dy_o. \tag{2.32}$$

The respective variables are as follows:

$$A = \hat{k}_x^- = \sin\theta_1 - \sin\theta_2\cos\theta_3$$

$$B = \hat{k}_y^- = -\sin\theta_2\sin\theta_3$$

$$C = \hat{k}_z^- = -(\cos\theta_1 + \cos\theta_2)$$

$$a = (\hat{k}^+ - R_o\hat{k}^-)_x = \sin\theta_1(1 - R_o) + \sin\theta_2\cos\theta_3(1 + R_o)$$

$$b = (\hat{k}^+ - R_o\hat{k}^-)_y = \sin\theta_2\sin\theta_3(1 - R_o)$$

$$c = (\hat{k}^+ - R_o\hat{k}^-)_z = \cos\theta_2(1 + R_o) - \cos\theta_1(1 - R_o) \tag{2.33}$$

An additional step can remove the gradient expressions by the use of integration by parts and assuming the area of integration extends over $-X \leq x_o \leq X$ and $-Y \leq y_o \leq Y$. The scattering function now has the form:

$$\psi^{sc}(\mathbf{r}) = \frac{-ie^{ikr}}{4\pi r}(\frac{Aa}{C} + \frac{Bb}{C} + c)\int_{S_M} e^{ik\phi(x_o,y_o)}dx_o dy_o + \psi_e \tag{2.34}$$

where the edge effects, ψ_e,

$$\psi_e = \frac{-ke^{ikr}}{4\pi r}[\frac{ia}{kC}\int(e^{ik\phi(X,y_o)} - e^{ik\phi(-X,y_o)})dy_o$$

$$+\frac{ib}{kC}\int(e^{ik\phi(x_o,Y)} - e^{ik\phi(x_o,-Y)})dx_o], \tag{2.35}$$

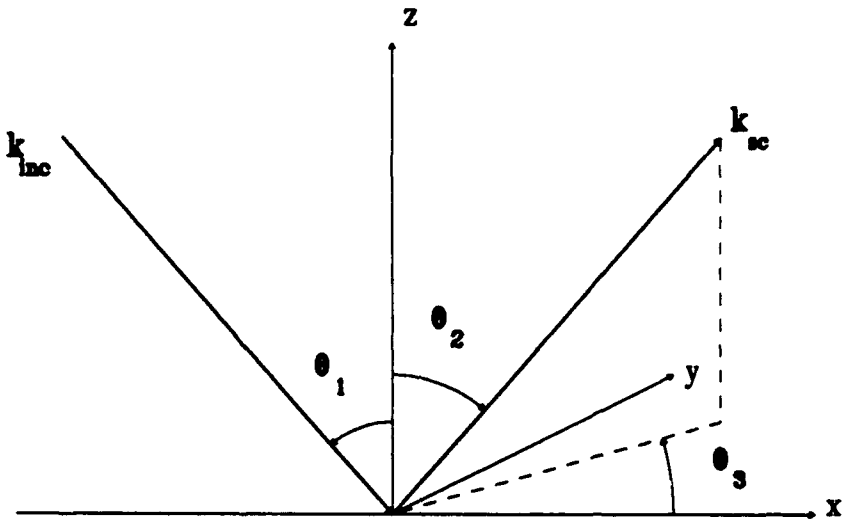

Figure 2.2. Coordinate system.

and the phase is

$$\phi(x_o, y_o) = Ax_o + By_o + Ch(x_o, y_o). \tag{2.36}$$

The angular terms can be consolidated into one function

$$F(\theta_1, \theta_2, \theta_3) = \frac{1}{2}(\frac{Aa}{C} + \frac{Bb}{C} + c), \tag{2.37}$$

yielding (Ogilvy, 1991, p. 83)

$$\psi^{sc}(\mathrm{r}) = \frac{-ie^{ikr}}{4\pi r} 2F(\theta_1, \theta_2, \theta_3) \int_{S_M} e^{ik\phi(x_o, y_o)} dx_o dy_o + \psi_e \tag{2.38}$$

The edge effects are often assumed negligible. While this is a convenient assumption it is only true for the near specular case ($\theta_1 = \theta_2$, $\theta_3 = 0$) (Ogilvy, 1991, p. 85).

This formula is the general form and now can be used to look at the coherent and diffuse fields. With the introduction of the probability density function the average mean acoustic field becomes

$$\langle \psi^{sc} \rangle = \frac{-ie^{ikr}}{4\pi r} 2F(\theta_1, \theta_2, \theta_3) \int_{S_M} \int_{-\infty}^{\infty} e^{ik\phi(x_o, y_o)} p(h) dh dx_o dy_o + \langle \psi_e \rangle \quad (2.39)$$

The amplitude of the coherent field, after some manipulation, becomes

$$\langle \psi^{sc} \rangle = \frac{-ie^{ikr}}{4\pi r} A_M c \chi(kC) (\frac{\sin kAX}{kAX})(\frac{\sin kBY}{kBY}) = \chi(kC)\psi_o^{sc} \quad (2.40)$$

where ψ_o^{sc} is the field scattered from a smooth surface of area A_M. For a Gaussian distribution the characteristic function is

$$\chi(s) = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} exp(-\frac{h^2}{2\sigma^2}) e^{ish} dh = e^{-s^2\sigma^2/2}. \quad (2.41)$$

From this equation 2.40 becomes

$$\langle \psi^{sc} \rangle = \chi(kC)\psi_o^{sc} = \psi_o^{sc} e^{-g^2/2} \quad (2.42)$$

where

$$g = k^2 C^2 \sigma^2 \quad (2.43)$$

For specular reflection the Rayleigh parameter, R_a, is equivalent to $\frac{1}{2}\sqrt{g}$.

For the diffuse field the mean amplitude would sum to zero but the average intensity, $\langle I_d \rangle$, can be evaluated.

$$\langle I_d \rangle = \langle \psi^{sc} \overline{\psi}^{sc} \rangle - \langle \psi^{sc} \rangle \langle \overline{\psi}^{sc} \rangle \quad (2.44)$$

Three additional assumptions are made about the surface. The first is that the dimensions of the surface are larger than the correlation length. The second is that the surface of interest is isotropic in order to use spherical coordinates.

The third is that the edge effects, $\langle \psi_e \rangle$, are not stochastic and give no contribution to the diffuse field ($h(\pm X, \pm Y) \equiv 0$)). From these assumptions the following is true (Ogilvy, 1991, p. 87)

$$\langle I_d \rangle = \frac{k^2 F^2}{2\pi r^2} A_M \int_0^\infty J_o(\sqrt{x^2 + y^2})$$

$$\times [\chi_2(kC, -kC, R) - \chi(kC)\overline{\chi_(kC)}] R dR. \tag{2.45}$$

From Ogilvy the two dimensional characteristic function is (1991, p. 89)

$$\chi_2(kC, -kC, R) = exp(-(k^2 C^2 \sigma^2 [1 - D(R)])) \tag{2.46}$$

and the surface correlation function $D(R)$ is

$$D(R) = e^{-R^2/\lambda_o^2}. \tag{2.47}$$

Finally this yields an expression for the diffuse field intensity of

$$\langle I_d \rangle = \frac{k^2 F^2 \lambda_o^2 e^{-g}}{2\pi r^2} A_M \sum_{n=1}^{\infty} \frac{g^n}{n! n} exp(-\frac{k^2 (A^2 + B^2) \lambda_o^2}{4n}) \tag{2.48}$$

This carries the restriction that $g < 1$ for the series to converge. The total field intensity is the sum of the coherent and diffuse fields.

$$\langle I \rangle = I_o e^{-g} + \langle I_d \rangle \tag{2.49}$$

The result of equation 2.49 is that fields of scattered energy from various Gaussian roughness can be predicted if the RMS height, wavevector, incident and scattering angles, illuminated area and correlation length are known. Ogilvy divides surfaces into three regimes for evaluation (1991, pp 91-93). The divisions are $g \ll 1$, for slightly rough surfaces, $g \sim 1$, for moderately rough surfaces, and $g \gg 1$, for very rough surfaces. The first and second regions

give information about the diffuse fields. The third region is dominated by the diffuse field and it becomes the total field. Additionally, in the second region the diffuse field is described as a range between a minimum and maximum possible.

For the slightly rough surface the diffuse field intensity is the first term in equation 2.48.

$$\langle I_d \rangle \sim \frac{k^2 F^2 \lambda_o^2}{4\pi r^2} g e^{-g} A_M exp(-\frac{k^2(A^2+B^2)\lambda_o^2}{4})$$

$$= \frac{k^4 C^2 F^2}{r^2} A_M P(kA, kB) \tag{2.50}$$

where $P(kA, kB)$ is equation 2.14.

For the moderately rough surface the range of diffuse intensity is in the range:

$$\frac{k^2 F^2 \lambda_o^2}{4\pi r^2} g e^{-g} A_M exp(-\frac{k^2(A^2+B^2)\lambda_o^2}{4}) \leq \langle I_d \rangle \sim\leq \frac{k^2 F^2 \lambda_o^2}{4\pi r^2} A_M \tag{2.51}$$

For very rough surfaces the total intensity magnitude has the following form:

$$\langle I \rangle = \frac{k^2 F^2 \sigma^2}{r^2} \frac{1}{g} A_M p_{12}(\frac{A}{C}, \frac{B}{C}) \tag{2.52}$$

where

$$p_{12}(x, y) = \frac{\lambda_o^2}{4\pi\sigma^2} exp(-\frac{\lambda_o^2(x^2+y^2)}{4\sigma^2}) \tag{2.53}$$

From equations 2.51, 2.52, and 2.53 some projections can be made about what level of backscatter will be seen given an RMS roughness and a correlation length.

Chapter 3

MATERIALS

In order to study the feasibility of imaging atherosclerotic plaque and quantitating its surface roughness suitable samples are needed that duplicate the surfaces seen in vivo. While real tissue samples can be utilized, it is more convenient, as well as more accurate, to use samples of a known roughness and correlation length. The references found in this research contain no data of RMS surface roughness or correlation length for typical lesions. Because data on the roughness of atherosclerotic plaque is only of an anecdotal nature, a wide range of roughness samples is required. The samples of roughness are in three forms. The first samples are Plexiglas blocks that have been sanded with various grades of sandpaper. The second samples are also Plexiglas blocks with glass beads or carbon particles bonded to their surfaces. The third type is a cast surface comparator from the Aluminum Association (300 19th Street, N.W., Washington, D.C.). In addition to the materials mentioned, there are three tissue samples used for demonstration purposes.

These samples provide an opportunity to test the scattering theory with real surfaces. There are some dissimilarities between the sample surfaces and blood vessel tissue however. The basic difference is that the materials have different density and sound speed than arterial tissue. This is discussed at the end of this chapter.

Sanded Surfaces

The Plexiglas blocks are 5 centimeter square surfaces, 2 centimeters thick. They are illustrated in Figure 3.1. The holes through the sides are for mounting. The Plexiglas blocks are sanded with the following sandpaper types: 60, 100, 180, 280, and 400 grit roughness. The sanding is accomplished by applying the paper to the surface in the same direction until the gloss of the smooth surface is replaced with a uniform sheen. This sheen is an indication that the abrasive has removed as much of the Plexiglas as is possible but before it has begun to actually remove the highest peaks.

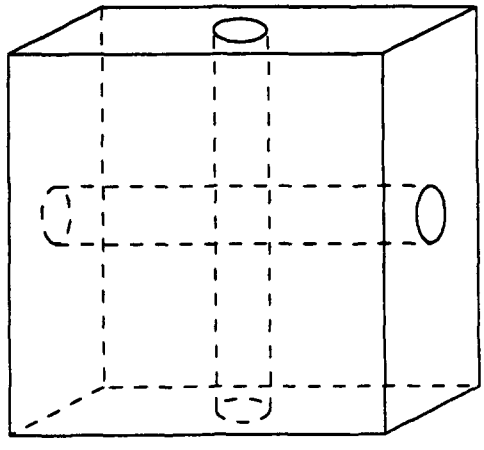

Figure 3.1. Plexiglas block.

Sandpaper is usually described by a grit number. While often the grit is assumed to represent the number of particles per linear inch, this is not the case. This number is a reference to the size of the sieve used to separate the abrasive particles. The grit number specifies the number of holes per linear inch in the sieve. The following Table 3.1 shows the actual sizes of the grains, and the associated grit number for closed coat sandpaper (Norton Co., 1990, p. 3). (Closed coat refers to the fact that the grains completely cover the backing paper. Open coat sandpaper has only 50% to 60% of the backing covered.) As can be seen in the table the sanded surface roughness is much less than the particle size of the sandpaper used to makes it. This is due to the distribution of the particles on the sandpaper. Adjacent particles prevent the full penetration into the surface of any given particle.

Table 3.1. Sandpaper information.

Grit Number	Particle Size μm	R_A μm	λ_o μm
60	268.0	3.5*	12.7*
100	141.0	2.0	9.5
180	78.0	1.2	7.5
280	44.0	0.85	5.0
400	23.6	0.58*	4.1*

* calculated

Three of the samples (100, 180, and 280 grit) were tested on a Talysurf 10 profilometer. The profilometer is only capable of measuring small roughness values. Anything above a value of $\sim 3.0\mu m$ would give inaccurate results and perhaps damage the machine. The profilometer moves a stylus over the surface and measures the height of the surface relative to a mean surface. The machine also makes a trace of the surface on a strip chart. A linear

regression for the roughness measured on the profilometer to the grain size of the sandpaper was made. The correlation for the regression is 0.999. The formula for the linear regression is:

$$R_A = 0.30137 + 0.01196d, \tag{3.1}$$

where d is particle size.

The correlation length for the sanded surfaces is found from the Talysurf 10 strip charts. Sixty-four points are sampled at $5\mu m$ from the chart and then correlated and the correlation length found. The measured values are then correlated with particle size, grit, and roughness. The correlation length had a correlation with particle size of -0.986, with grit of 0.993, and with roughness of -0.993. The values predicted from the grit correlation are used for the two surfaces not measured. The formula for the linear regression is:

$$\lambda_o = 2.609 + 0.025grit, \tag{3.2}$$

The predicted values are marked with an asterisk in Table 3.1.

In addition to the sanded surfaces, a block made with the 400 grit paper has five regions of extreme roughness of differing areas. The areas are 2, 3, 4, 5, and 10 millimeter squares. This is a resolution sample that is for maximum contrast. The roughened regions were made with a stylus and by physical inspection are rougher than any of the samples available. Unfortunately no RMS roughness or correlation length are known. This resolution sample is shown in Figure 3.2.

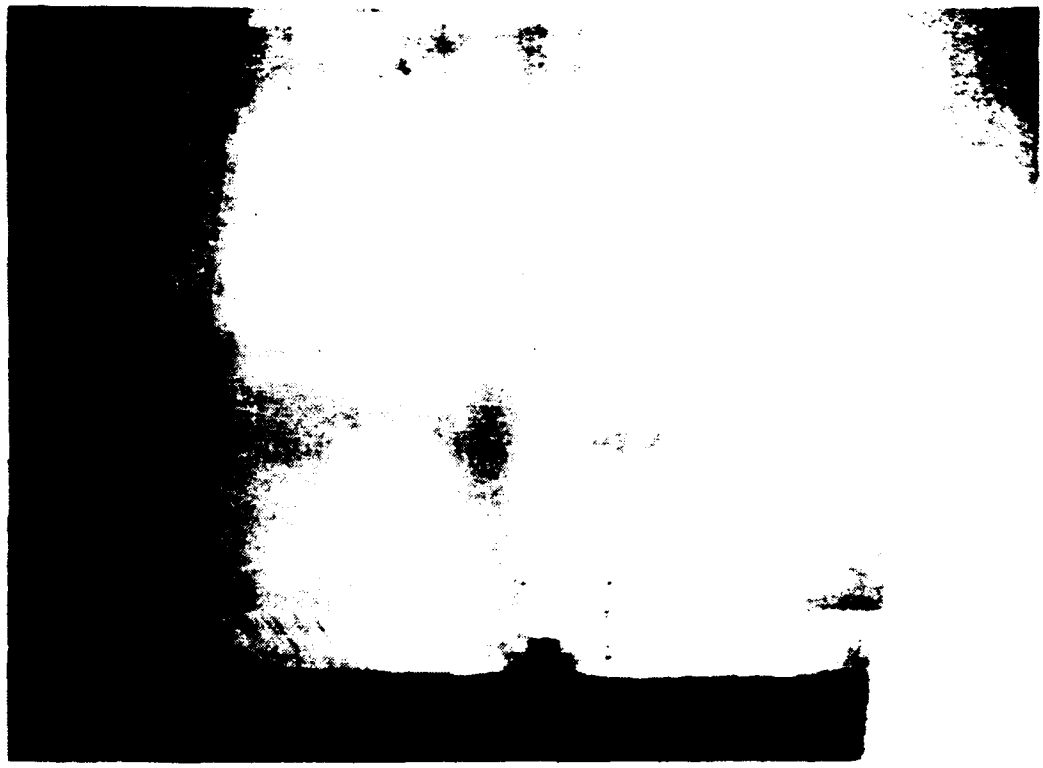

Figure 3.2. Resolution sample.

Glass Beads

The glass bead surfaces are prepared by placing glue on the portion of the Plexiglas for the beads. The beads are then spread over the area in sufficient quantity to assure complete coverage.

The glass bead surfaces required an analytical approach for finding the roughness because of the profilometer's sensitivity. The calculation of the roughness of a surface is based on the following formula:

$$R_A = \frac{h_1 + h_2 + h_3 \ldots h_n}{L} = \frac{1}{L}\int_0^L |h| dL \qquad (3.3)$$

The average roughness, R_A, is used generally for machined surfaces (Bennett and Mattsson, 1989, p.39). This formula implies that a horizontal midline exists where the area between the line of the surface trace above the midline equals the area of the surface trace below the midline and the midline. With a few assumptions the roughness can be calculated for the beaded surfaces.

The first assumption is that the beads are spherical and of the same diameter. While there is some variation among a group of beads, they are reasonably uniform. The second assumption is that the beads on the surface are distributed as densely as possible. This is possible with the uniformity of the spheres. The next assumption is that the glue used to apply the beads to the surface fills the volume below the spheres to their equators. This means the surface is composed of hemispheres. The next assumption is that the nominal roughness can be calculated by taking an average of two trace directions. Because of the dense packing of the spheres there is a direction where the spheres will be touching at their equators. Also because of the

uniformity of the spheres this is a straight line. Because of the symmetry of the arrangement this alignment will exist at 60° increments around a central bead, as in Figure 3.3. As the angle sweeps between the hexagonal peaks a maximum of roughness is found at the midpoint between the 60° spokes. A line taken anywhere else will result in a value between these two values. The average is a nominal value. It is not the exact value, but should yield a reasonable value considering the other assumptions.

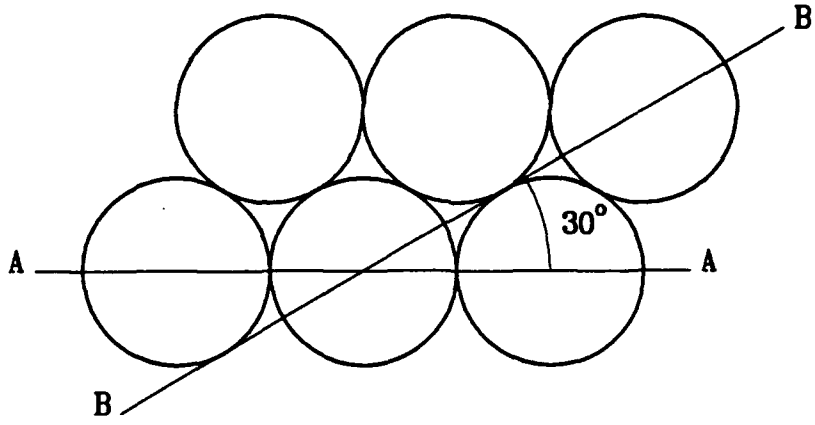

Figure 3.3. Beaded surface from above.

The direction where the spheres touch will be referred to as $A - A$, as in Figure 3.4. The other line, 30° away, will be referred to as $B - B$, as in Figure 3.5. Both lines are composed of repeating segments.

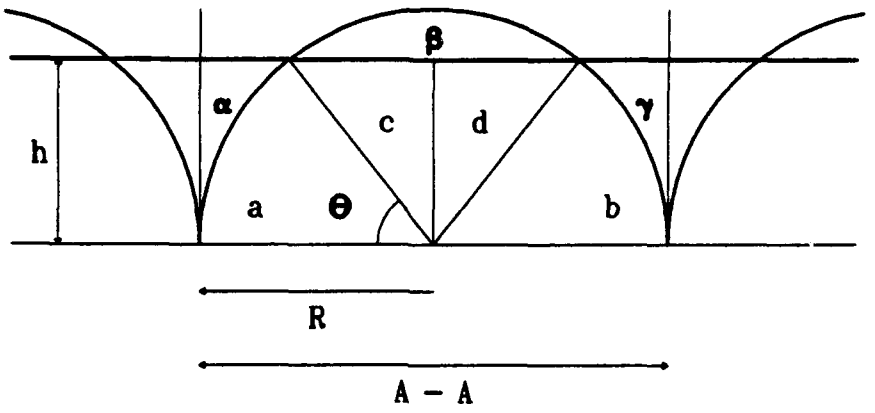

Figure 3.4. Beaded surface line $A - A$.

For the $A - A$ line, the segment length is the same as the diameter of the sphere, $2R$. The roughness is calculated by summing the areas above and below a midline and then dividing by the segment length. The height of the midline, h, is found by equating the areas above and below. The area of the semicircle is $\pi R^2/2$. The areas of concern, α, β, and γ, are described as the area below h and above the semicircle, the area above h but below the semicircle, and the area below h after the semicircle drops below h, respectively. Areas α and γ are the same size. To satisfy equation 3.3 the following is true:

$$\alpha + \gamma = \beta, \quad \alpha = \gamma \tag{3.4}$$

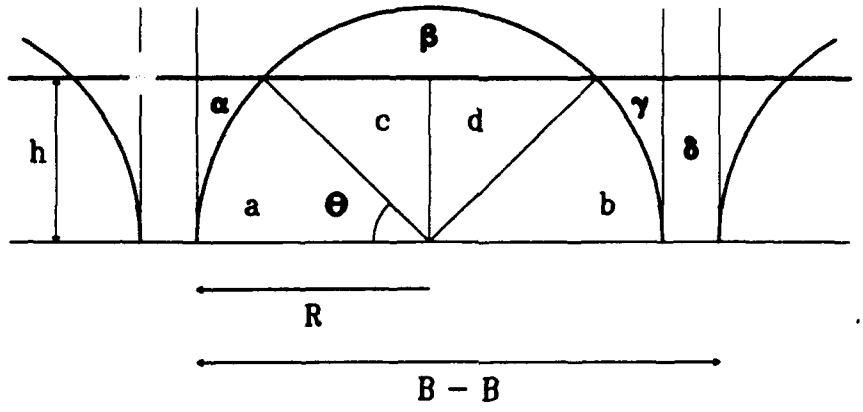

Figure 3.5. Beaded surface line $B - B$.

The area of β is the difference between the total area of the semicircle minus the area below h inside the semicircle:

$$\beta = \pi R^2/2 - (a + b + c + d), \quad a = b, \quad c = d \tag{3.5}$$

The figures show that θ describes the angle between the base of the semicircle and the intersection of h and the semicircle. From this we have the following area formulas:

$$a = \theta R^2/2, \quad \theta \; in \; radians \tag{3.6}$$

$$b = \frac{1}{2} R \cos\theta R \sin\theta \tag{3.7}$$

$$\beta = R^2(\frac{\pi}{2} - \theta - \sin\theta\cos\theta) \tag{3.8}$$

$$\alpha = \gamma = R^2(\sin\theta - \theta/2 - cos\theta sin\theta/2) \tag{3.9}$$

Solving for the angle θ:

$$\theta = \arcsin\frac{\pi}{4} \tag{3.10}$$

Therefore for line segment $A - A$ the roughness, R_A is:

$$R_A = \frac{1}{2R}R^2(\pi/2 + 2\sin\theta - 2\theta - 2\sin\theta\cos\theta) = 0.1813R \tag{3.11}$$

For the $B - B$ line the repeating distance includes the space between two spheres that are oriented at a 30° angle to the $A - A$ line. This additional area below h is labelled δ and is summed with α and γ.

$$\alpha + \gamma + \delta = \beta, \quad \alpha = \gamma \tag{3.12}$$

The distance can be found easily from geometry. The distance between the centers of spheres on the $B - B$ line is $2\sqrt{3}R$. Therefore δ is defined by:

$$\delta = h(\sqrt{3} - 1)R = 2\sin\theta(\sqrt{3} - 1)R^2 \tag{3.13}$$

From this formula comes a new value for θ.

$$\theta = \arcsin\frac{\pi}{4\sqrt{3}} \tag{3.14}$$

This gives a new value for the roughness along the $B - B$ line.

$$R_A = \frac{1}{2R\sqrt{3}}R^2(\pi/2 + 2\sqrt{3}\sin\theta - 2\theta - 2\sin\theta\cos\theta) = 0.4018R \tag{3.15}$$

Averaging the two values gives the roughness as $0.2916R$. Figure 3.6 shows a plot of R_A as a function of the spacing between beads. Both the roughness and spacing are expressed as fractions of the radius.

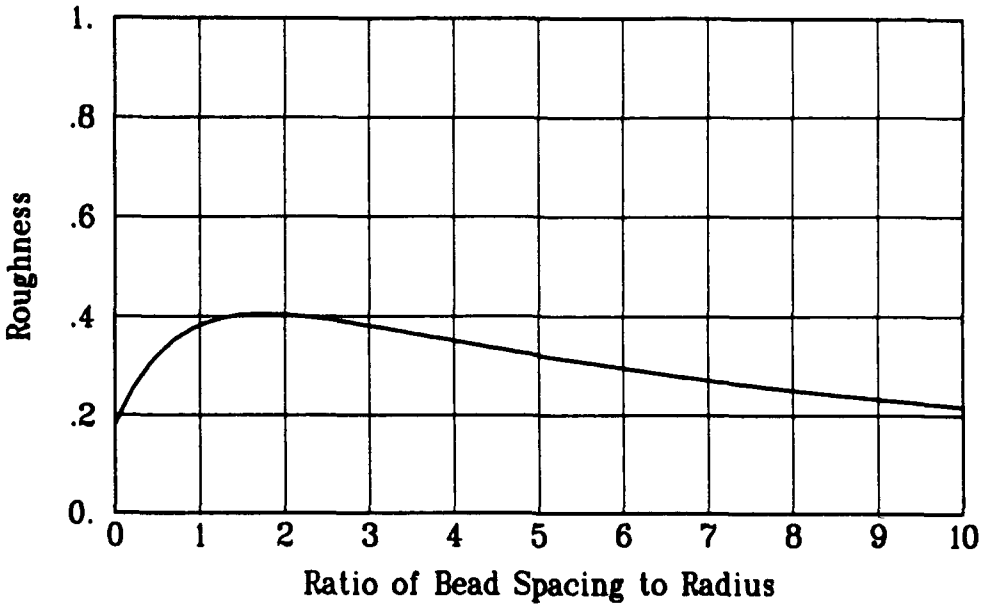

Figure 3.6. R_A as function of bead spacing.

For the sake of argument calculating the case of no glue at all, h is above the substrate by an increased distance R, δ is increased by $2R^2(\sqrt{3}-1)$.

$$\delta = 2(1+sin\theta)(\sqrt{3}-1)R^2 \tag{3.16}$$

$$\theta = sin^{-1}(\frac{1}{\sqrt{3}}(\frac{\pi}{4}+1-\sqrt{3})) \tag{3.17}$$

This case yields then a roughness:

$$R_A = \frac{1}{2R\sqrt{3}}R^2(\pi/2+2(\sqrt{3}-1)+2\sin\theta$$

$$+2(\sqrt{3}-1)\sin\theta-2\theta-2\sin\theta\cos\theta) = 0.8713R \tag{3.18}$$

These results show that a dense pack set of spheres of uniform size will have a maximum roughness associated with some multiple of the radius that is less than one.

Correlation length for the bead surface can also be calculated. Figure 3.7 shows a function depicting the bead surface and the correlation function. There are 32 points per bead profile and the correlation drops below e^{-1} at the tenth point. The correlation length is equal to $5/16R$.

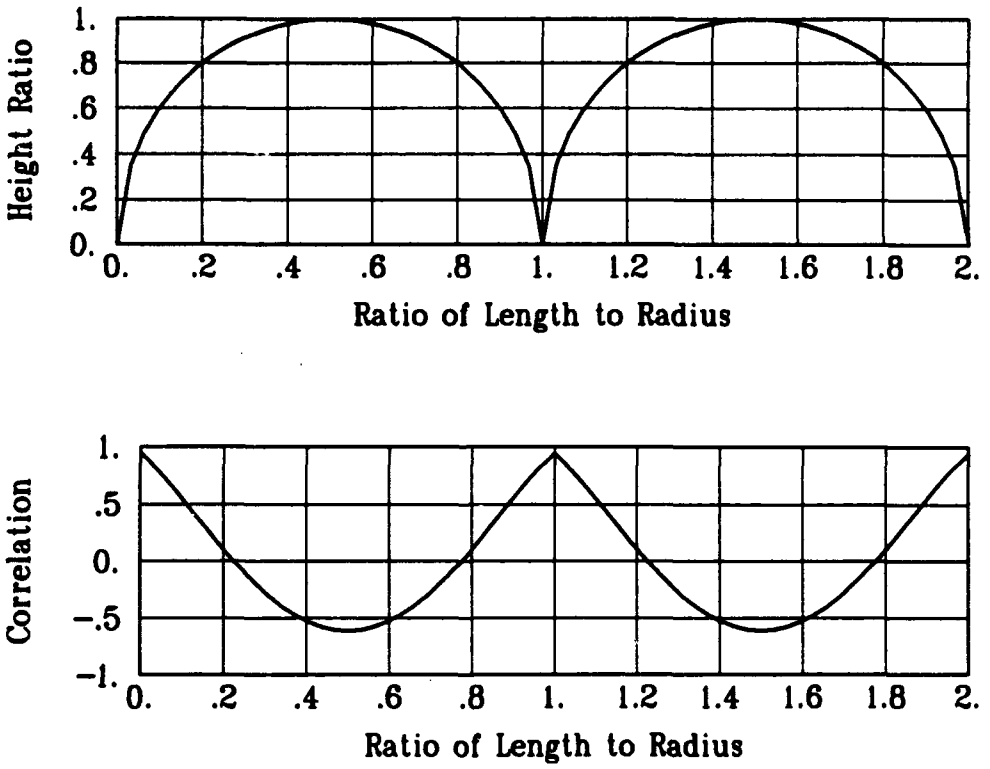

Figure 3.7. Bead profile and correlation function.

Two bead surfaces have been prepared and used. The first surface has the largest bead size covering the entire surface. The surface distribution is several layers deep due to too much glue in the application of the beads

which allowed the beads to become more than one layer deep. The second sample surface for beads is a Plexiglas block with 5 millimeter wide vertical strips composed of five different sizes of spheres. The pattern ($237\mu m$, $50\mu m$, $105\mu m$, $150\mu m$, $215\mu m$) is repeated twice for the full width of the block. The data for the beaded surfaces are displayed in Table 3.2 and the surface is shown in Figure 3.8.

Table 3.2. Glass bead information.

Dia. μm	Range μm	R_A μm	λ_o μm
50	40 - 70	7.25	7.81
105	85 - 120	15.23	16.41
150	140 - 160	21.75	23.44
215	205 - 225	31.18	33.59
237	177 - 297	34.37	37.03

Cast Surface Comparator

The cast surface comparator is shown in Figure 3.9. This surface is used as a standard for surface roughness in the aluminum casting industry. The surface is divided into nine areas with specified roughness in microinches. Table 3.3 shows the conversion to units of micrometers. The table also lists a nominal correlation length that is 5.4 times the R_A value. This is the average of the measured sanded surfaces compared to their R_A values. While this is probably not accurate it does provide a nominal value. Contact with the Aluminum Association Inc. indicated that they had no data on the correlation

Figure 3.8. Beaded strip surface.

length associated with their standard. The surfaces are far too rough for the Talysurf profilometer.

Table 3.3. Cast surface information.

R_A μin	R_A μm	λ_o μm
20	0.51	2.75
60	1.53	8.26
120	3.05	16.47
200	5.08	27.43
300	7.62	41.15
420	10.67	57.62
560	14.22	76.79
720	18.29	98.77
900	22.86	123.44

Tissue Samples

Three samples of artery tissue have also been studied. The samples are intended as demonstrations only. Because the vessel has to be slit and laid flat the character of the tissue is basically altered. The tissue samples shown in Figures 3.9 and 3.10 demonstrate how difficult it was to maintain a reasonably flat surface to scan. Figure 3.10 shows tissue that is relatively normal with only a small portion near the edge with some calcification. Figure 3.11 shows an extreme case of calcified lesion over the entire surface.

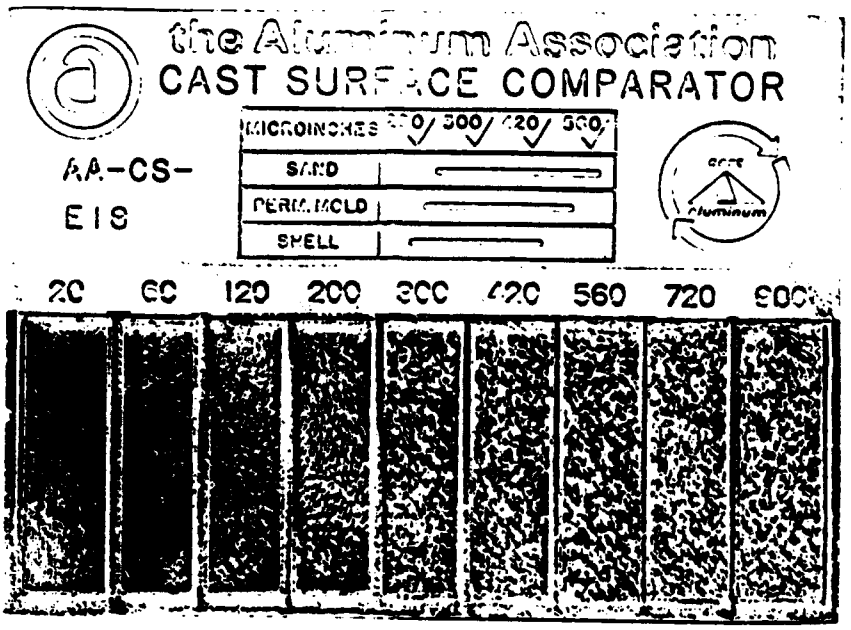

Figure 3.9. Cast surface comparator.

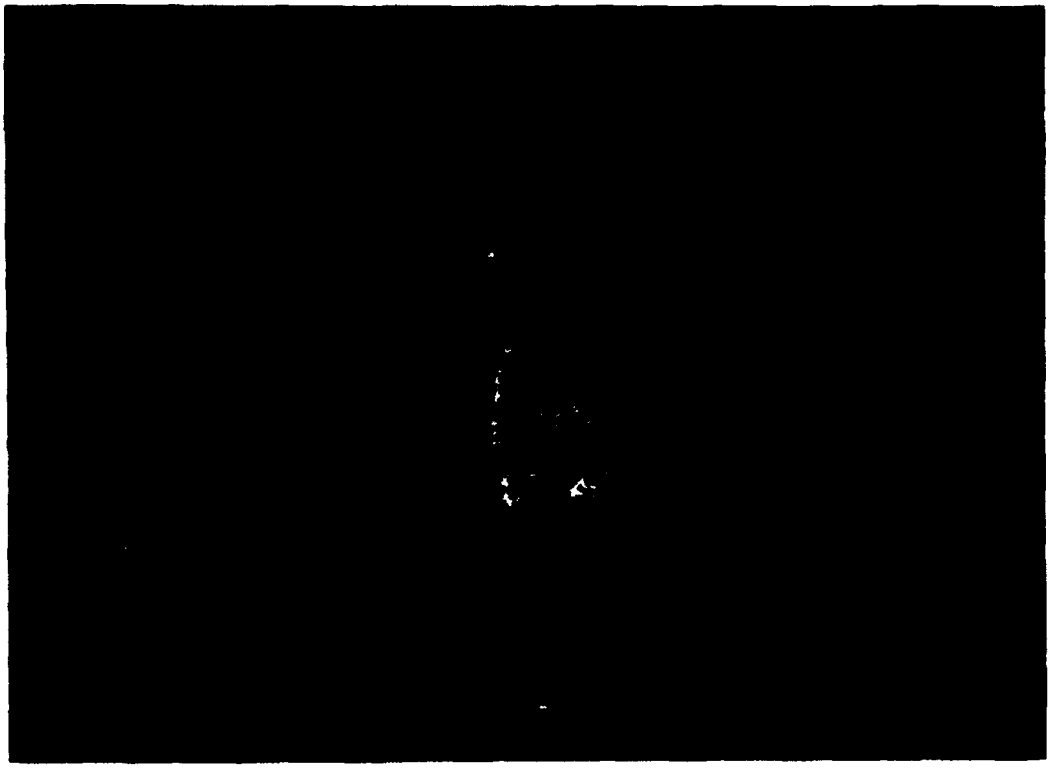

Figure 3.10. Nearly normal tissue.

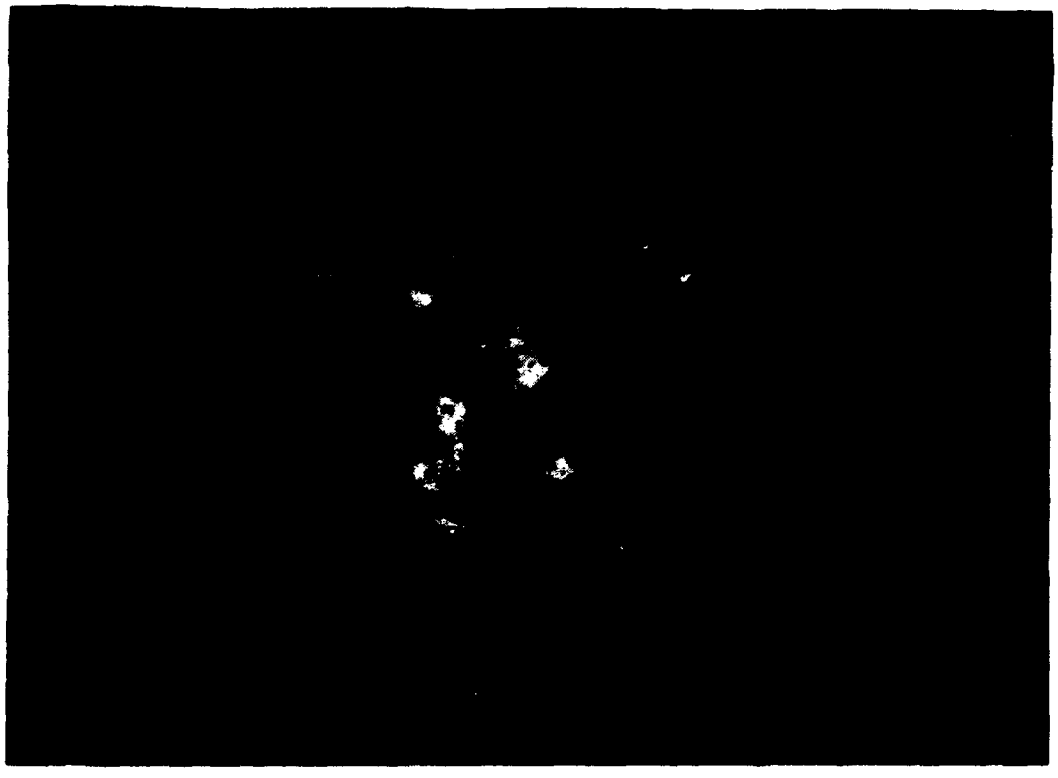

Figure 3.11. Calcified tissue.

The third sample does not have a photograph. The sample had a rather unusual structure of two tubules, perhaps blood vessels or scar tissue that formed around some surgical device, that crossed the main vessel section. This portion was removed when the vessel was slit for scanning. The rest of the vessel is relatively normal.

Reflection Coefficient

With the use of any substitute for the real material some factors can not be duplicated. This section covers the differences in material constants that affect the reflection of acoustic energy from a surface. The reflection and transmission of waves from an interface are determined by the material characteristics of the two media.

The theoretical basis for reflection would classify the reflection from the vessel wall to be more like a liquid-liquid interface than a liquid-solid interface. The major constituent of most biological tissue is water. This is the case for both blood and blood vessels. The sample surfaces, however, are all solids. The reflection from them will vary depending on their physical attributes of ρ, density, c, sound speed; both longitudinal (c_L) and shear (c_T), and these in turn depend on β, the bulk modulus and G,the shear modulus. Table 3.4 lists the relevant values (Kinsler et al., 1982, pp. 461-462)(Shung, 1985, p. 310).

The reflection and transmission coefficients for liquid-liquid interfaces are developed in the following manner (Brekhovskikh, 1960, pp. 16-18) (Kinsler et al., 1982, pp. 131-133). The geometry is illustrated in Figure 3.12.

Table 3.4. Material constants.

Material	Sound Speed m/s	Density kg/m^3	Impedance $(\cdot 10^{-6})$ $kg/m^2 s$	Bulk Modulus $(\cdot 10^{-10})$ Pa	Shear Modulus $(\cdot 10^{-10})$ $Pa \cdot s/m$
Water	1481	998	1.48	0.218	-
Aorta	1570	1051	1.65	-	-
Aluminum	6300	2700	17.0	7.5	2.4
Glass	5600	2300	12.9	3.9	2.5
Lucite	2650	1200	3.2	0.65	0.14

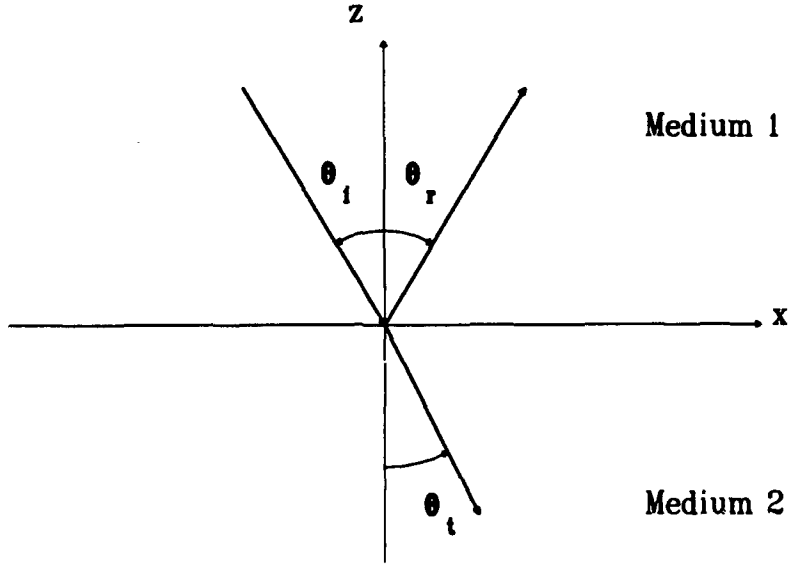

Figure 3.12. Reflection geometry.

The incident, reflected and transmitted pressures at the interface are defined as:

$$\mathbf{p}_i = \mathbf{P}_i e^{i(\omega t - k_1 x \cos\theta_i + k_1 x \sin\theta_i)}$$

$$\mathbf{p}_r = \mathbf{P}_r e^{i(\omega t + k_1 x \cos\theta_r + k_1 x \sin\theta_r)}$$

$$\mathbf{p}_t = \mathbf{P}_t e^{i(\omega t - k_2 x \cos\theta_t + k_2 x \sin\theta_t)} \tag{3.19}$$

The transmission angle is assumed complex initially. The boundary condition of pressure continuity at the interface requires that

$$\mathbf{P}_i e^{i(\omega t - k_1 x \sin\theta_i)} + \mathbf{P}_r e^{i(\omega t + k_1 x \sin\theta_r)} = \mathbf{P}_t e^{i(\omega t - k_2 x \sin\theta_t)} \tag{3.20}$$

For this to be true for all x, $\theta_i = \theta_r$. It also follows that with constant frequency,

$$\frac{\sin\theta_i}{c_1} = \frac{\sin\theta_t}{c_2} \tag{3.21}$$

which is the familiar Snell's law. The magnitudes of the functions mean that

$$1 + \mathbf{R} = \mathbf{T}, \tag{3.22}$$

where the reflection ratio is defined as $\mathbf{R} = \mathbf{P}_r/\mathbf{P}_i$ and the transmission ratio is defined as $\mathbf{R} = \mathbf{P}_t/\mathbf{P}_i$. The second boundary condition requires that particle velocity normal to the surface is zero. This results in a second equation in $\mathbf{R}$ and $\mathbf{T}$.

$$1 - \mathbf{R} = \frac{z_1 \cos\theta_t}{z_2 \cos\theta_i}\mathbf{T} \tag{3.23}$$

where $z_i = \rho_i c_i$, acoustic impedance. From equations 3.22 and 3.23,

$$\mathbf{R} = \frac{(z_2/z_1) - (\cos\theta_t/\cos\theta_i)}{(z_2/z_1) + (\cos\theta_t/\cos\theta_i)} \tag{3.24}$$

The value of θ_t can become complex if $c_1 < c_2$ as can be seen from

$$\cos\theta_t = \sqrt{1-\sin^2\theta_t} = \sqrt{1-(c_2/c_1)^2\sin^2\theta_i} \tag{3.25}$$

when $(c_2/c_1)^2\sin^2\theta_i > 1$. The incident angle at which this occurs is called the critical angle and is defined as

$$\sin\theta_c = \frac{c_1}{c_2} \tag{3.26}$$

While reflected waves are of primary concern for this thesis, the transmitted wave information is included for completeness. The transmitted pressure then becomes

$$\mathbf{p}_t = \mathbf{P}_t e^{-\gamma z} e^{i(\omega t - k_2 x \sin\theta_t)} \tag{3.27}$$

where

$$\gamma = k_2\sqrt{(c_2/c_1)^2\sin^2\theta_i - 1}$$

The transmitted wave travels along the x axis and decays away from the boundary. For θ_t real:

$$\mathbf{T} = \frac{2z_2/z_1}{(z_2/z_1) + (\cos\theta_t/\cos\theta_i)} \tag{3.28}$$

For θ_t imaginary, $\mathbf{T} = 0$. The reflection power level, $(20\log(\mathbf{R}))$, as a function of incidence angle is plotted in Figure 3.13.

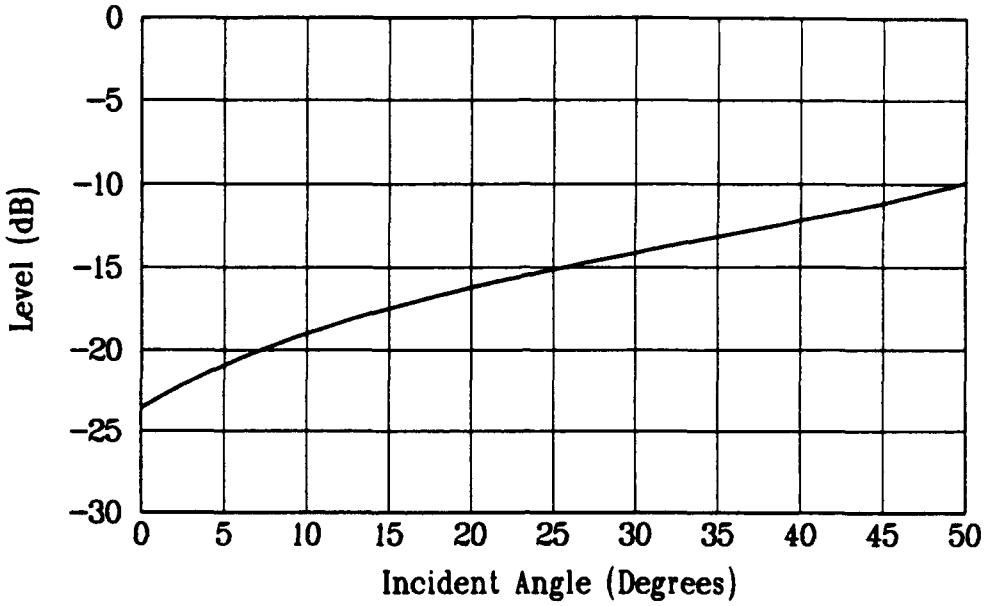

Figure 3.13. Reflection power level for aorta tissue.

The theoretical basis for the liquid-solid interface is similar in development. The result is somewhat more complicated by the addition of shear waves in the solid (Brekhovskikh, 1960, pp. 28-35). The particle velocity is defined as

$$\mathbf{v} = grad\phi + curl\psi. \tag{3.29}$$

where ϕ is the longitudinal potential and ψ is the transverse (shear) potential. The potential ψ is chosen so that only its y component is non-zero, since only the x and z components will contribute due to the orientation selected. The velocity components are as follows.

$$v_x = \frac{\partial\phi}{\partial x} - \frac{\partial\psi}{\partial z}, \quad v_y = 0, \quad v_z = \frac{\partial\phi}{\partial z} + \frac{\partial\psi}{\partial x} \tag{3.30}$$

The associated wave equations are:

$$\nabla^2\phi = \frac{1}{c_L^2}\frac{\partial^2\phi}{\partial t^2}, \quad \nabla^2\psi = \frac{1}{c_T^2}\frac{\partial^2\psi}{\partial t^2} \tag{3.31}$$

where c_L and c_T are the longitudinal and shear velocities. The velocities can be expressed in terms of Lame's parameters ξ and μ or equivalently in terms of the bulk modulus, β and the shear modulus, G.

$$c_L = \sqrt{\frac{\xi + 2\mu}{\rho}} \quad c_T = \sqrt{\frac{\mu}{\rho}} \tag{3.32}$$

$$c_L = \sqrt{\frac{\beta + \frac{4}{3}G}{\rho}} \quad c_T = \sqrt{\frac{G}{\rho}} \tag{3.33}$$

Note that the shear velocity is always less than the longitudinal velocity. The stress tensor components for this geometry are:

$$Z_x = \lambda(\frac{\partial u_x}{\partial x} + \frac{\partial u_z}{\partial z}) + 2\mu\frac{\partial u_x}{\partial z},$$

$$Z_y = 0, \quad Z_z = \mu(\frac{\partial u_x}{\partial z} + \frac{\partial u_z}{\partial x}) \tag{3.34}$$

For the liquid medium $\psi = 0$ and $\mu = 0$. The boundary conditions are continuity of Z_z:

$$\lambda_1\nabla^2\phi_1 = \lambda_2\nabla^2\phi_2 + 2\mu_2(\frac{\partial^2\phi_2}{\partial z^2} + \frac{\partial^2\psi_2}{\partial x\partial z});$$

Z_x equal to zero:

$$2\frac{\partial^2\phi_1}{\partial x\partial z} + \frac{\partial^2\psi_2}{\partial x^2} - \frac{\partial^2\psi_2}{\partial z^2} = 0;$$

and continuity of u_z:

$$\frac{\partial\phi_1}{\partial z} = \frac{\partial\phi_2}{\partial z} + \frac{\partial\psi_2}{\partial x}. \tag{3.35}$$

The sound wave is defined by the following formulas for the incident wave, ϕ_i, the reflected wave, ϕ_r, the transmitted longitudinal wave, ϕ_2, and the transmitted transverse wave, ψ_2.

$$\phi_i = Ae^{i(\omega t - k_1 x\cos\theta_i + k_1 x\sin\theta_i)} \tag{3.36}$$

$$\phi_r = ARe^{i(\omega t + k_1 x\cos\theta_r + k_1 x\sin\theta_r)} \tag{3.37}$$

$$\phi_2 = ATe^{i(\omega t - k_2 x\cos\theta_t + k_2 x\sin\theta_t)} \tag{3.38}$$

$$\psi_2 = ASe^{i(\omega t - \kappa_2 x\cos\gamma_t + \kappa_2 x\sin\gamma_t)} \tag{3.39}$$

The wave numbers are k_1, k_2, and κ_2. From the boundary conditions the following relationships are found for the reflection and transmission coefficients using

$$k_1\sin\theta_i = k_2\sin\theta_t = \kappa_2\sin\gamma_t \tag{3.40}$$

and

$$Z_1 = \frac{\rho_1 c_1}{\cos\theta_i} \quad Z_2 = \frac{\rho_2 c_{L2}}{\cos\theta_t} \quad Z_t = \frac{\rho_2 c_{T2}}{\cos\gamma_t} \tag{3.41}$$

yields

$$R = \frac{Z_2\cos^2(2\gamma_t) + Z_t\sin^2(2\gamma_t) - Z_1}{Z_2\cos^2(2\gamma_t) + Z_t\sin^2(2\gamma_t) + Z_1} \tag{3.42}$$

$$T = \frac{2Z_2\cos(2\gamma_t)}{Z_2\cos^2(2\gamma_t) + Z_t\sin^2(2\gamma_t) + Z_1} \tag{3.42}$$

$$S = \frac{2Z_t\sin(2\gamma_t)}{Z_2\cos^2(2\gamma_t) + Z_t\sin^2(2\gamma_t) + Z_1} \tag{3.42}$$

Given these coefficients there are three regimes of operation. When $0 < \sin\theta_i < c_1/c_{L2}$, both θ_t and γ_t will be real angles and the reflection coefficient will be real. When $c_1/c_{L2} < \sin\theta_i < c_1/c_{T2}$, γ_t will be real, θ_t will be complex, and the reflection coefficient will be complex. The longitudinal transmitted

wave will travel along the boundary in the solid and the shear wave will be a plane wave. When $c_1/c_{T2} < \sin\theta_i$, both θ_t and γ_t will be complex angles and the reflection coefficient will be unity. Both transmitted waves will propagate along the boundary. Table 3.5 documents the critical angles, intensity loss at 0 degrees and 15 degrees for each of the materials and aorta tissue. For the three materials in use the plots for their reflection power level, $(20\log(\mathbf{R}))$, are shown in Figures 3.14, 3.15, and 3.16 for aluminum, glass, and lucite, respectively.

Table 3.5. Critical angles and intensity loss.

Material	Critical Angles °	Intensity Loss(dB) 0°	Intensity Loss(dB) 15°
Aorta	-	-21.6	-15.0
Aluminum	13.6,29.8	-1.5	-1.5
Glass	15.3,26.7	-2.0	-1.8
Lucite	34.1	-8.8	-8.9

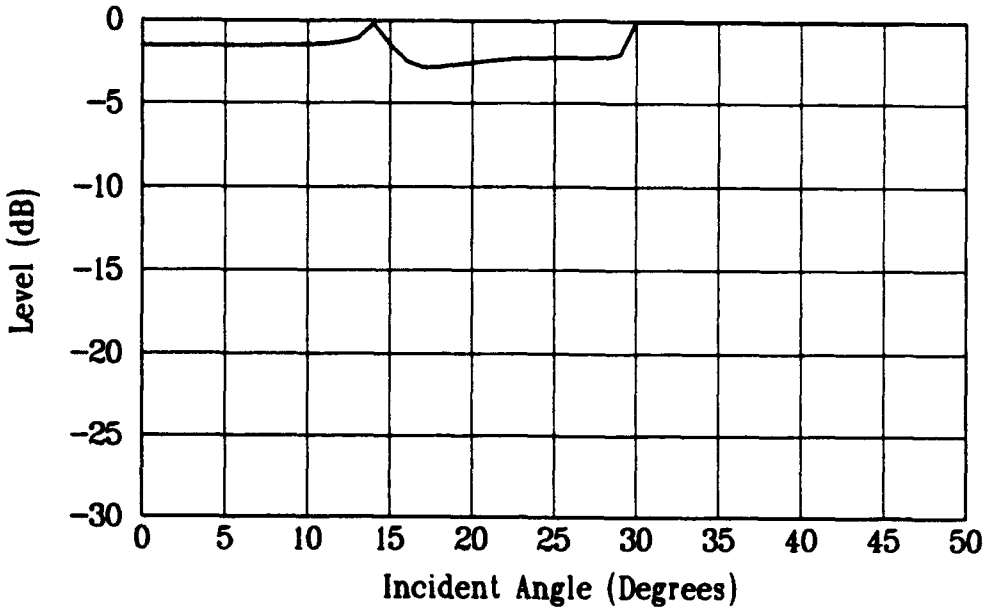

Figure 3.14. Reflection power level for aluminum.

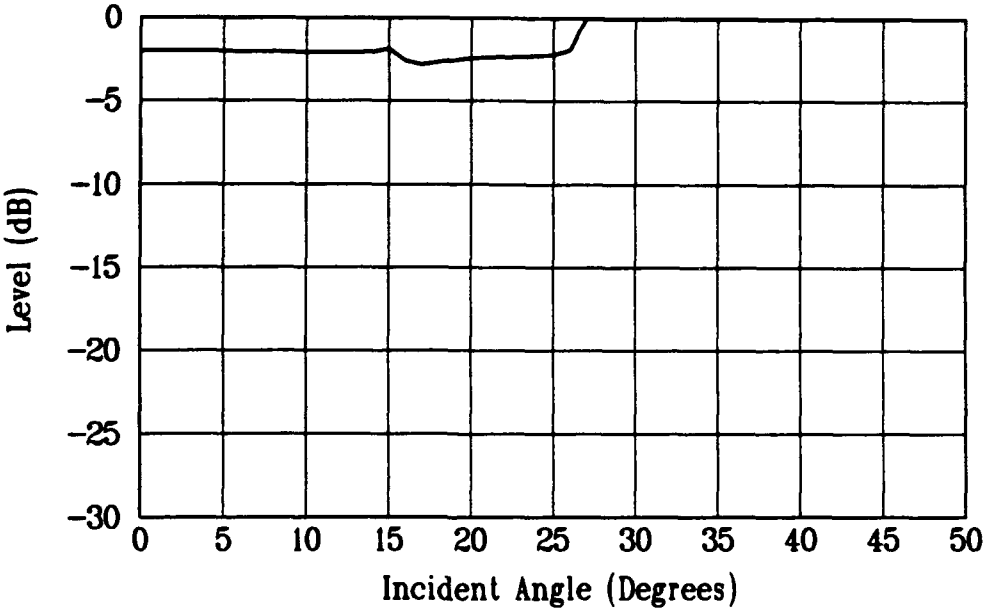

Figure 3.15. Reflection power level for glass.

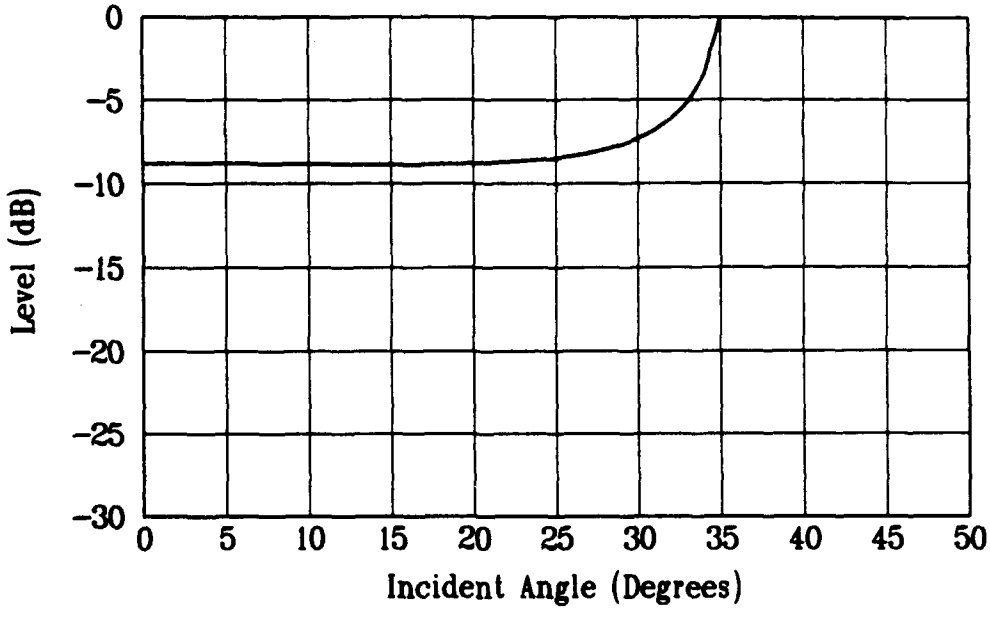

Figure 3.16. Reflection power level for lucite.

Chapter 4

METHODS

The experiments for this research were conducted in the Ultrasound Laboratory of the Bioengineering Department. The experiments consist of two types. The first experiment type is used to plot scattering level from various surfaces as the angle of incidence is varied. This is used to establish a baseline angle for investigating the backscatter out of the main specular lobe. The second type of experiment examines a surface at two angles. The first angle is at 0 degrees from the vertical to the mean plane of the surface. The second angle is at 15 degrees from the vertical. This chapter discusses the equipment used and the procedure followed.

Experiment Type 1

The purpose of the first experiment was to find the amount of scattered energy returned from a smooth surface, various sanded surfaces and one glass bead surface over a range of incident angles. The system has a monostatic transducer. This means that the transducer used for the transmission of the pulse is also used as the receiver. As the sample is rotated the incident and reflection angles are equal.

The procedure for this experiment is to mount the plexiglas block on its adaptor and place into the shaft of the angular measurement vernier. The

sanded plexiglas surfaces are oriented so that their parallel sanded directions are perpendicular to the incident beam. This is insures that the beam will see the roughest surface as the sample is rotated. The shaft is the nearest vertical shaft in the figure. The vernier is the squat round object the shaft runs through. The sample is submerged below the vernier. The second vertical shaft has the transducer mounted on it. The beam from the transducer is aligned to get the maximum return from a stationary object on the vernier shaft. After this alignment the pulse is bounced off the surface of a sample. The sample is also adjusted to get the maximum return level at specular reflection. Once a peak is found the angle is recorded for use as the starting point.

The first surface is the smooth plexiglas sample for use as a baseline standard. The signal return is routed to the oscilloscope and the peak level is aligned with a point on the screen. Because the level at specular reflection is very high it is attenuated through a series of calibrated adjustable attenuators. As the angle is changed to a new value the attenuators are set to adjust the received level back to the established peak value on the oscilloscope.

In this manner each surface is tested and the results recorded. For each the starting setting is the same as for the smooth surface. In this manner all surfaces can be compared. For the glass bead surface, because the return is very diffuse, the smooth reverse side is set up and then the vernier is rotated 180° to arrive at 0° for the beads.

For this first experiment set the following equipment is used. The description of each piece is included for completeness and for questions of repeatability. The transducer is driven by a 7.5 Megahertz sine wave signal generated with a Tektronix FG504 Function Generator. The signal is converted to a

pulse with a Tektronix PG505 Pulse Generator. The pulse is 5 microseconds long and has a pulse repetition rate of 1 kilohertz. The rise and fall rates of the pulse are 1 microsecond. The attenuators are Kay Elemetrics Corporation Models 1/432D and 432D. The Oscilloscope is a Tektronix SC 502. The flow chart of the apparatus for this experiment is shown in Figure 4.1.

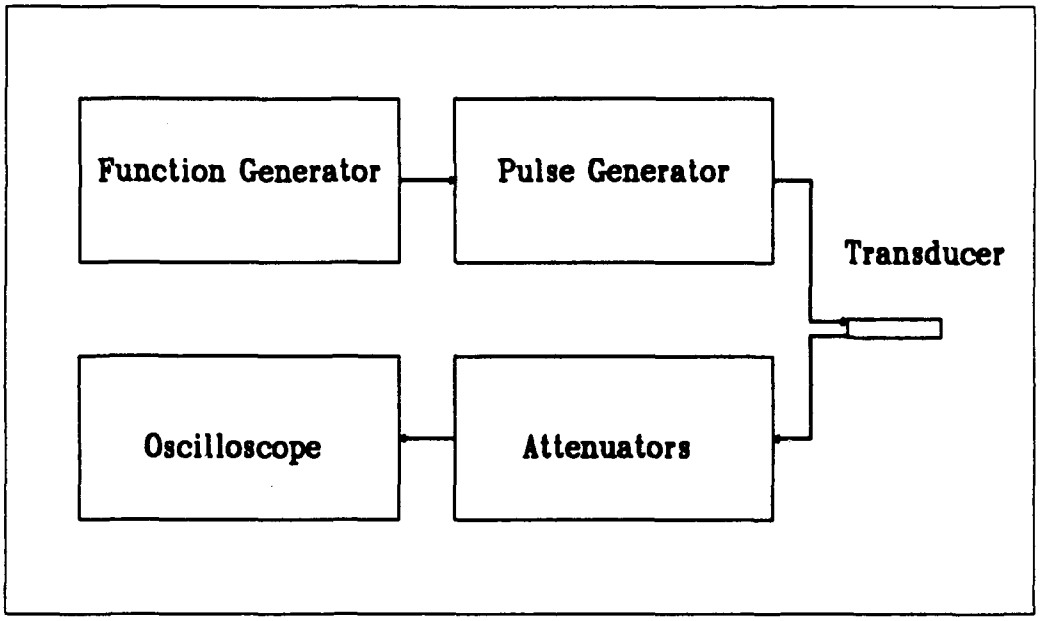

Figure 4.1. Equipment flow chart for first experiment.

Experiment Type 2

The second experimental type is designed to make complete C-mode scans of the sample surfaces at two angles, 0° and 15° . A C-mode scan has the azimuth position displayed on the horizontal axis and the elevation position displayed on the vertical axis (IEEE Standard Radar Definitions, 1982, p. 9). The view is the same as viewing the sample with one's eye. The angles chosen are to demonstrate that detail lost in the specular (0°) scan can be recovered in an oblique (15°) scan. Because the scan rates of the device are very slow the data is stored in a computer file for both later viewing and data analysis. The majority of the scans use a PDP-11 computer with a Peritek monitor for displaying the scans. The scan made of the aluminum cast comparator is done with a Compaq 386.

For the second series of experiments the following procedures were used. The samples were set in position in the same manner as before for the 0° scan. The scan is controlled by the computer. The computer now controls the position of the transducer, by the use of three stepper motors. The stepper motors can be seen in Figure 4.1 as the three cylindrical objects oriented in the three axes directions of x (transverse), y (vertical), and z (longitudinal) from the alignment for the transducer. The transducer is set in position manually for its starting position. The program starts and the computer then moves the transducer in a linear raster pattern to completely scan a region of the sample. The width and height of the scan are variables for the program.

The level is stored in a file and also displayed on the screen. The step size for the PDP-11 scans was set to $100\mu m$. The Peritek monitor has a 512×512

pixel display. This can display a region as large as the samples. The scans of this size unfortunately can take up to two days to complete. The PC scans are also designed to be 512 points wide. Because the aluminum cast comparator is 11 centimeters wide compared to the 5 centimeter width of the blocks, a larger step size of $204\mu m$ is required for coverage. Before a scan is conducted a preview scan is conducted to find the highest likely level to be encountered. This allows an attenuation level to be set to keep the dynamic range of the surface within the stored level range of 0 to 255.

Once the scan for 0° is made the transducer returns to its starting position. The sample is rotated to 15° and the transducer is adjusted in x and z to compensate for the change in relative position to the sample. A second preview scan is made to find the range of values before a full second scan is done.

For the second series of experiments the following equipment was used. The transducer is a Sound Technologies transducer with a resonance frequency of 10 megahertz. The equipment is also shown in Figure 4.2. A KB Aerotech UTA-4 combines many of the functions of several devices into one. This device now controls the pulse level, pulse repetition rate, gated return signal and attenuation level. A Sony/Tektronix 390AD receives the return signal from the KB Aerotech and digitizes it into 1024 points of $\pm$ voltage selected on the 390AD at a rate of 60 megahertz. The digitized values are then sent to the computer which takes the absolute value of the voltage and shifts the value right by one bit to arrive at the data range of 0 to 255. The flow chart of the apparatus for this experiment is shown in Figure 4.2.

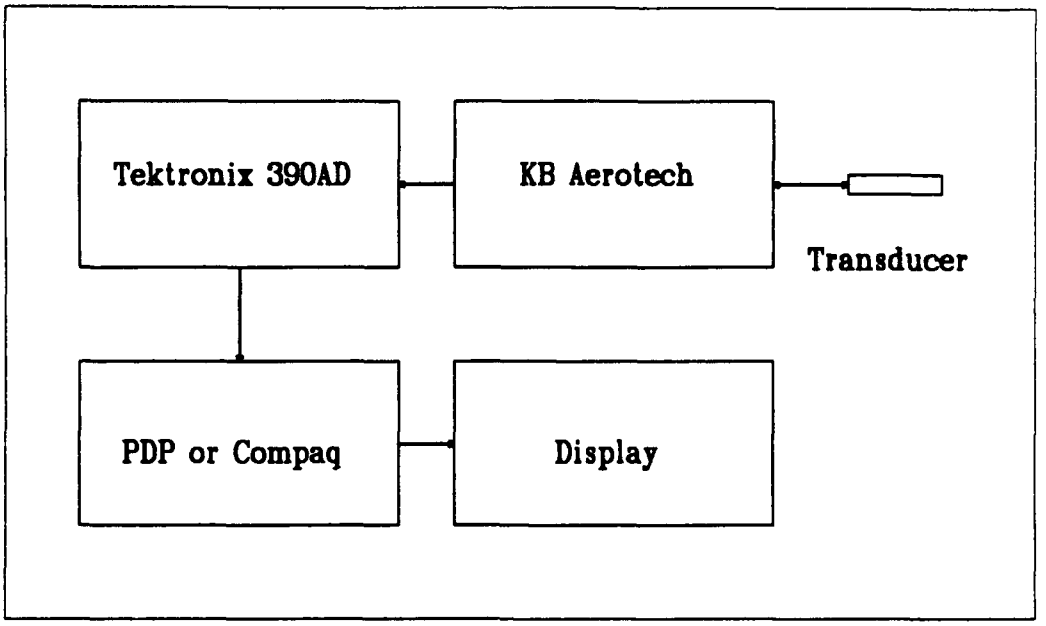

Figure 4.2. Equipment flow chart for second experiment.

Chapter 5

RESULTS

This chapter presents the results of the experiments discussed in Chapter 4 with the application of the theory presented in Chapter 2. The additional factor of differences in reflection coefficients presented in Chapter 3 is also considered. The first results to discuss are for the first type of experiment. Next the results for the second experiment set are shown. The second set results are presented in two forms. The first is the qualitative results of the plots or C-mode images of the scans. The second form is the statistical nature of the levels and how that relates to the theoretical levels expected. Finally there is a presentation of the conclusions of this thesis, along with some suggestions for future research.

First Experiment Results

The first set of experiments show the level of the backscattered energy for a smooth surface,the four sanded surfaces, and a beaded surface. The levels are tabulated in Table 5.1 and are referenced to the reflected level at normal incidence for the smooth surface. The angles are rounded to the nearest whole angle.

Table 5.1. Backscattering levels (-dB) for sandpapered surfaces.

Angle °	Smooth	Beads	60 Grit	100 Grit	180 Grit	280 Grit	400 Grit
0	0.0	22.9	0.2	3.0	1.3	0.6	0.1
1	2.6	18.0	4.8	3.3	3.2	4.4	0.8
2	4.1	27.6	13.2	9.1	10.6	18.9	7.6
3	32.9	19.8	24.8	36.5	29.4	34.6	26.9
4	35.8	25.2	30.0	38.0	39.6	40.7	33.6
5	40.6	23.4	28.1	39.6	42.0	41.6	39.6
6	41.0	28.3	27.0	41.0	41.8	41.3	40.6
7	42.4	20.6	30.0	39.6	45.4	44.9	41.0
8	46.7	24.6	29.8	39.4	46.6	50.2	45.8
9	45.7	21.9	32.0	41.6	46.3	46.6	45.8
10	46.3	27.9	30.9	40.9	49.0	47.6	45.6
11	49.4	25.4	30.2	41.4	57.6	50.9	47.6
12	56.6	23.4	30.9	40.6	63.6	60.7	51.9
13	63.6	22.3	30.0	39.6	63.6	63.6	59.7
14	63.6	23.1	29.0	39.3	63.6	63.6	63.6
15	63.6	21.6	30.0	39.6	63.6	63.6	63.6
16	63.6	20.1	31.8	39.6	63.6	63.6	63.6
17	63.6	20.6	31.4	39.6	63.6	63.6	63.6
18	63.6	23.3	31.0	39.7	63.6	63.6	63.6
19	63.6	22.4	29.0	40.4	63.6	63.6	63.6
20	63.6	22.3	28.5	41.3	63.6	63.6	63.6
21	63.6	21.2	28.2	42.6	63.6	63.6	63.6
22	63.6	20.7	28.3	44.1	63.6	63.6	63.6
23	63.6	21.1	28.0	44.6	63.6	63.6	63.6
24	63.6	23.3	27.8	44.6	63.6	63.6	63.6
25	63.6	24.5	28.0	44.7	63.6	63.6	63.6
26	63.6	23.4	28.3	44.9	63.6	63.6	63.6
27	63.6	22.3	29.0	45.1	63.6	63.6	63.6
28	63.6	21.5	30.0	45.0	63.6	63.6	63.6
29	63.6	21.5	31.0	44.7	63.6	63.6	63.6
30	63.6	21.4	31.9	45.2	63.6	63.6	63.6
31	63.6	20.7	32.1	44.8	63.6	63.6	63.6
32	63.6	20.5	31.0	41.8	63.6	63.6	63.6
33	63.6	20.6	27.9	38.7	63.6	63.6	63.6
34	63.6	21.3	27.0	37.4	63.6	63.6	63.6
35	63.6	21.4	28.0	37.6	63.6	63.6	63.6
36	63.6	21.3	27.7	37.8	63.6	63.6	63.6
37	63.6	22.6	27.6	37.6	63.6	63.6	63.6
38	63.6	23.3	27.0	36.7	63.6	63.6	63.6
39	63.6	23.0	26.1	36.4	63.6	63.6	63.6
40	63.6	21.9	25.7	36.1	63.6	63.6	63.6

Table 5.1. (continued)

Angle °	Smooth	Beads	60 Grit	100 Grit	180 Grit	280 Grit	400 Grit
41	63.6	21.2	25.0	36.2	63.6	63.6	63.6
42	63.6	20.6	24.0	37.6	63.6	63.6	63.6
43	63.6	20.6	25.0	36.4	63.6	63.6	63.6
44	63.6	21.3	25.0	36.6	63.6	63.6	63.6
45	63.6	22.0	25.0	37.4	63.6	63.6	63.6

The value for I_o in the calculated examples uses the actual values from the measurements for the smooth surface sample. Figure 5.1 shows the smooth surface backscatter referenced to the specular maximum.

Now that an established I_o exists the equations 2.49, 2.50, 2.51, and 2.52 can be evaluated and compared to the observed levels. Each surface is compared in the following plots. Each pair of plots show the observed levels in the first plot and the predicted levels are shown in the next plot. Figure 5.2 shows the scattered levels from the 400 grit sanded surface. Figure 5.3 shows the predicted scattered levels from the 400 grit using equations 2.49 and 2.50 which are appropriate for $g = 0.0024$. Figures 5.4 through 5.11 show results for 280, 180, 100, and 60 grit sanded surfaces and the predicted levels with g values of 0.0051, 0.0101, 0.0281, and 0.0860, respectively.

Figure 5.12 shows the observed levels from the $237\mu m$ diameter bead surface. Figure 5.13 shows the predicted scattering level range from the $237\mu m$ diameter bead surface using equations 2.49 and 2.51 for $g = 1.17$.

The plots for the sanded surfaces of 100 and 60 grit do not match the level of the predicted values, while the shape does show good agreement with the increased level in the vicinity of the critical angle. This would imply that

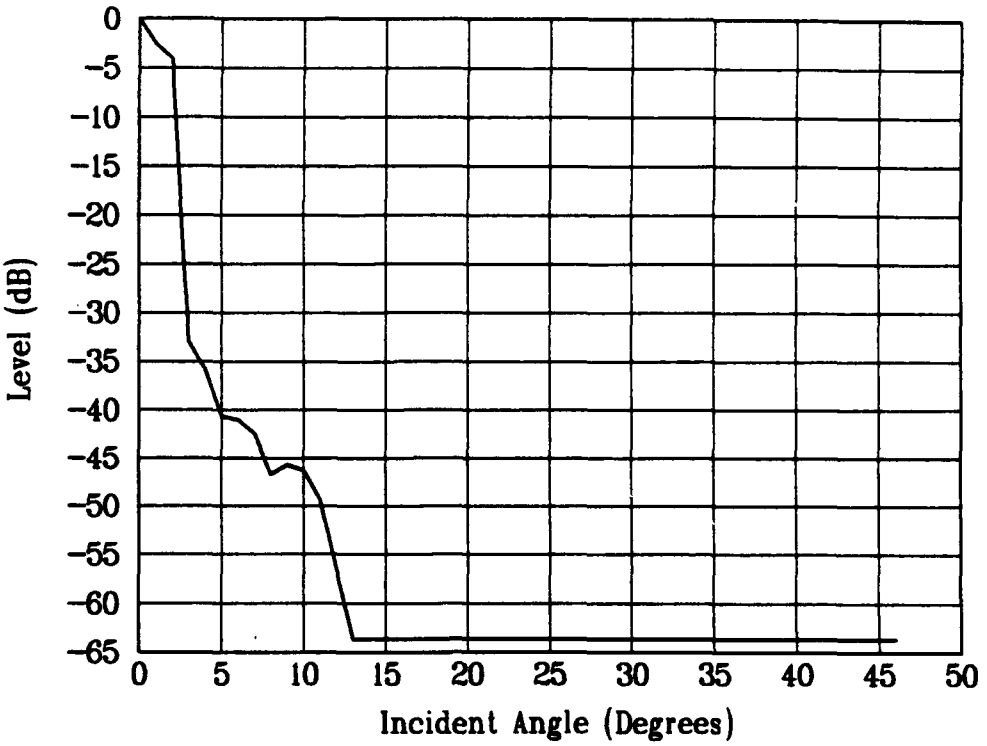

Figure 5.1. Smooth surface backscatter.

one of the variables in equation 2.50 is incorrect. The only one that has some question about it is the correlation length.

The correlation length required to match the level for the 100 grit sample to compensate for the 20 dB higher observed level would be $95\mu m$, which is 10 times the quantity found for the correlation length. The correlation length required to match the level for the 60 grit sample to compensate for the 25 dB higher observed level would be $226\mu m$, which is 17.8 times the quantity extrapolated for the correlation length. These values are approximately 55

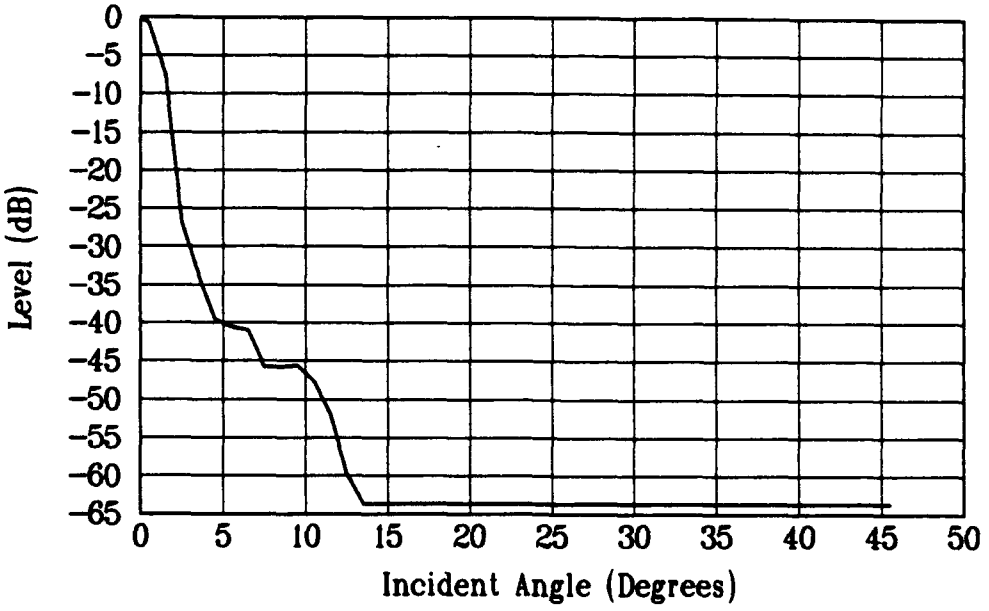

Figure 5.2. Actual levels from 400 sanded surface.

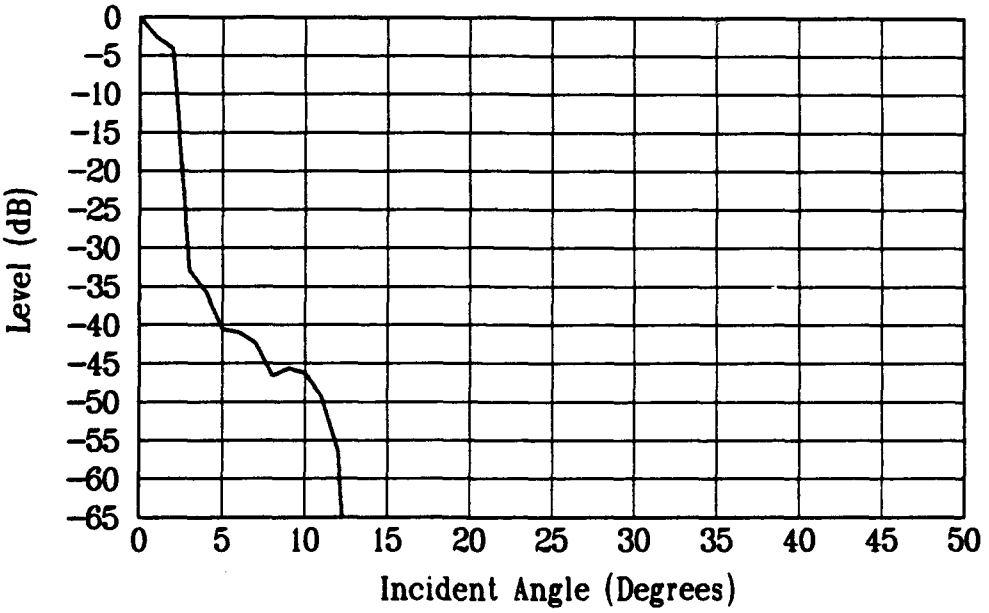

Figure 5.3. Predicted levels from 400 sanded surface.

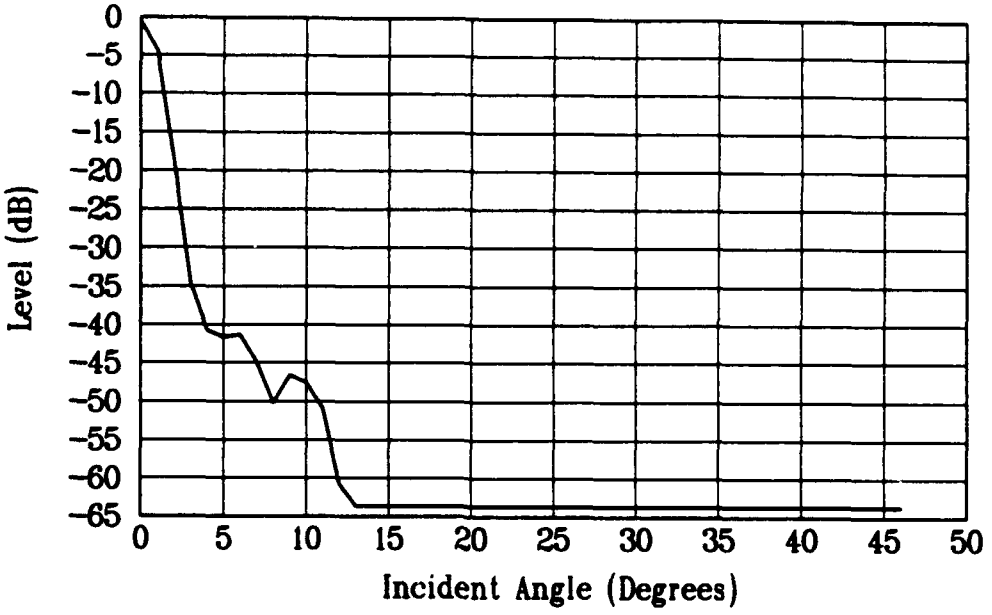

Figure 5.4. Actual levels from 280 sanded surface.

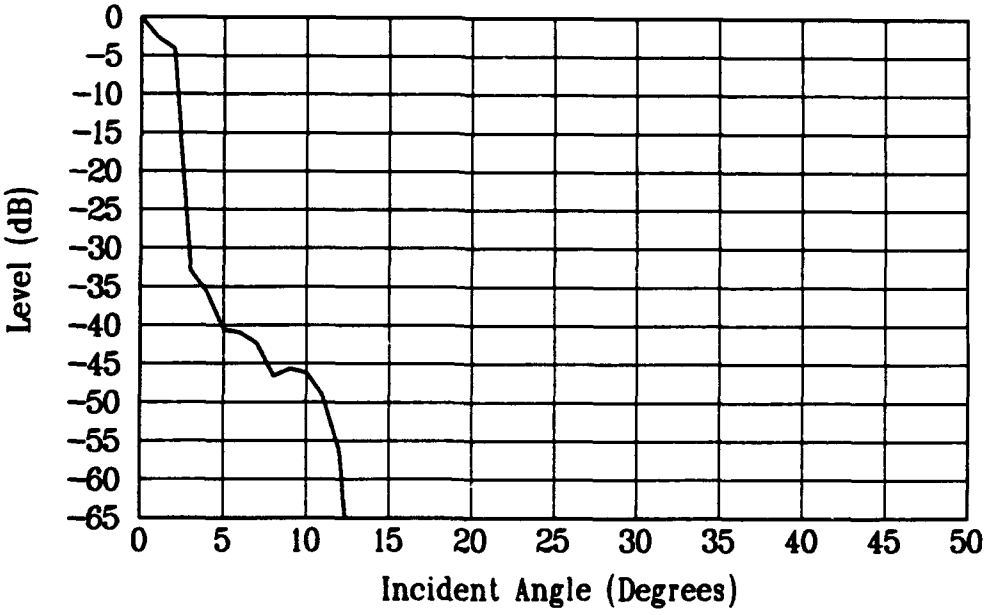

Figure 5.5. Predicted levels from 280 sanded surface.

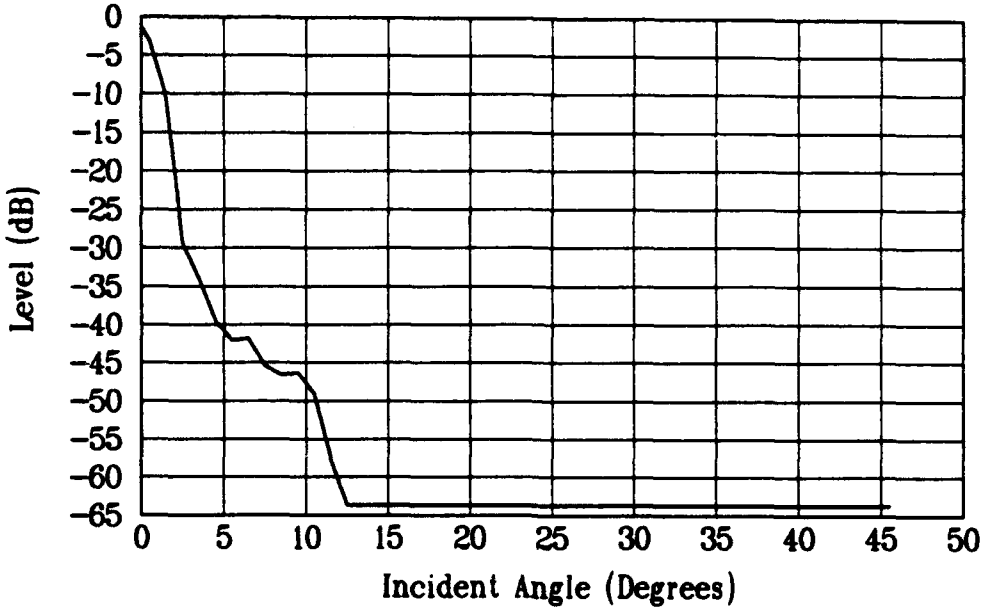

Figure 5.6. Actual levels from 180 sanded surface.

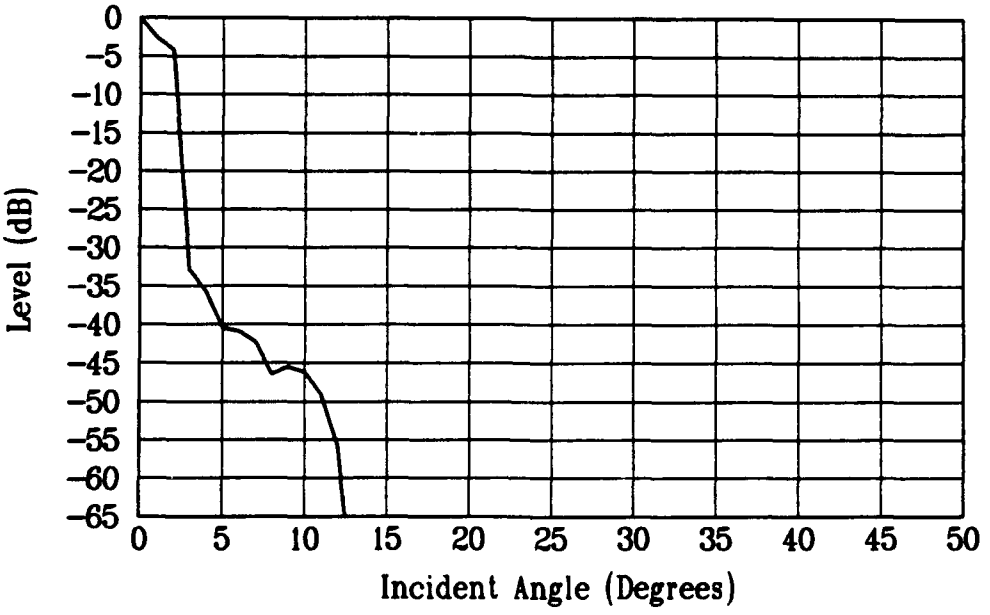

Figure 5.7. Predicted levels from 180 sanded surface.

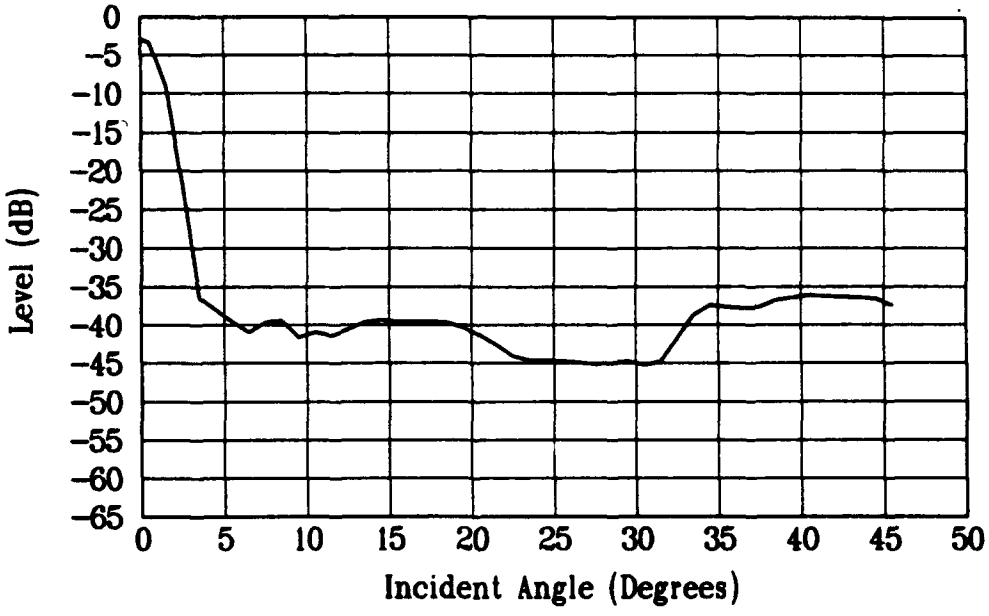

Figure 5.8. Actual levels from 100 sanded surface.

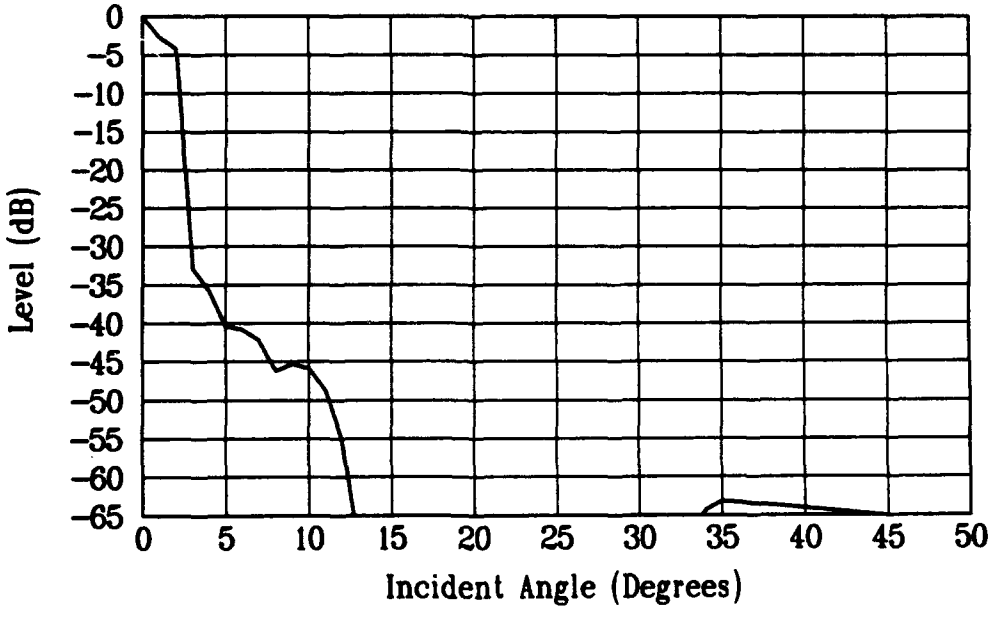

Figure 5.9. Predicted levels from 100 sanded surface.

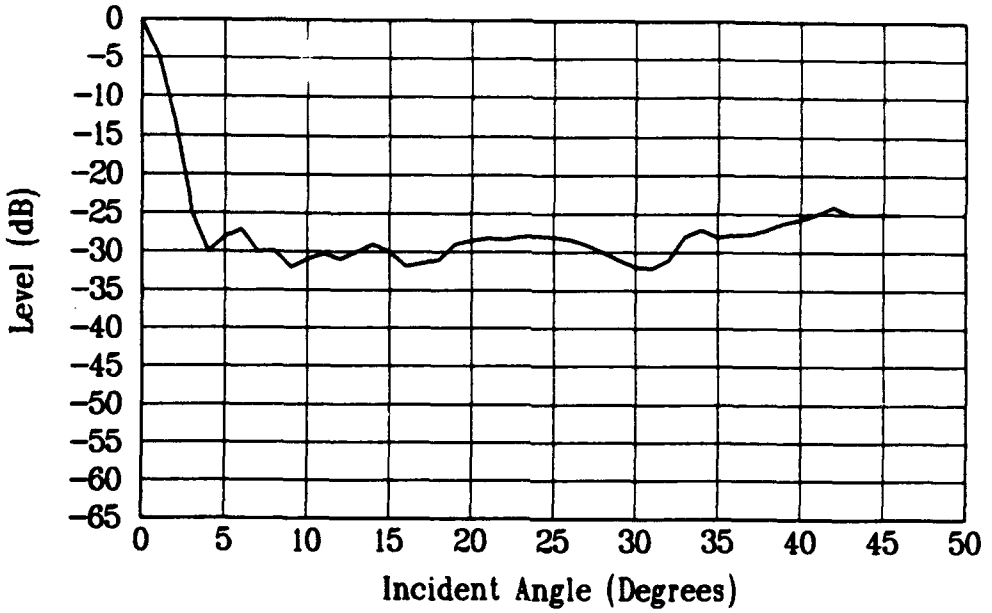

Figure 5.10. Actual levels from 60 sanded surface.

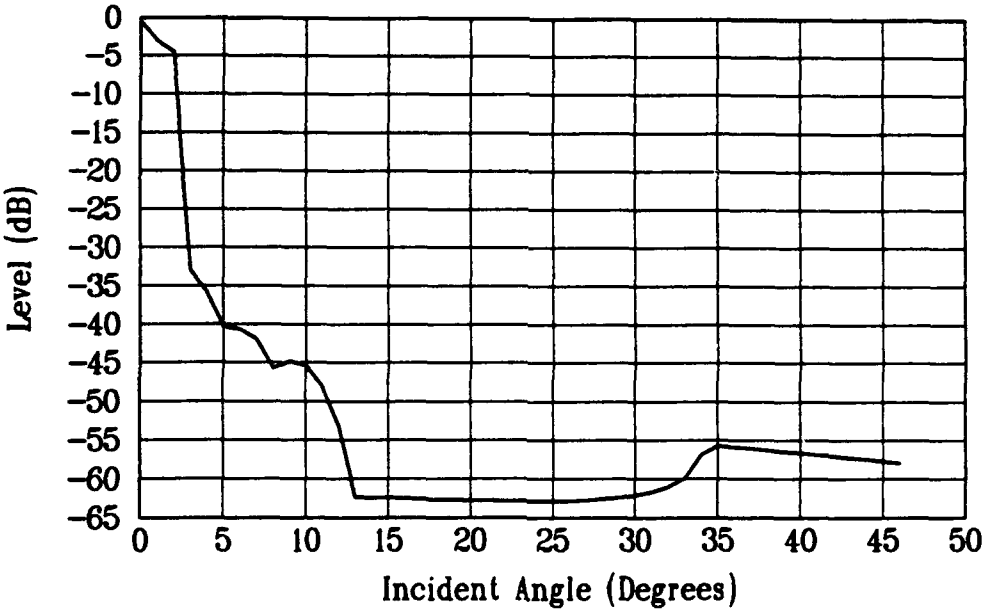

Figure 5.11. Predicted levels from 60 sanded surface.

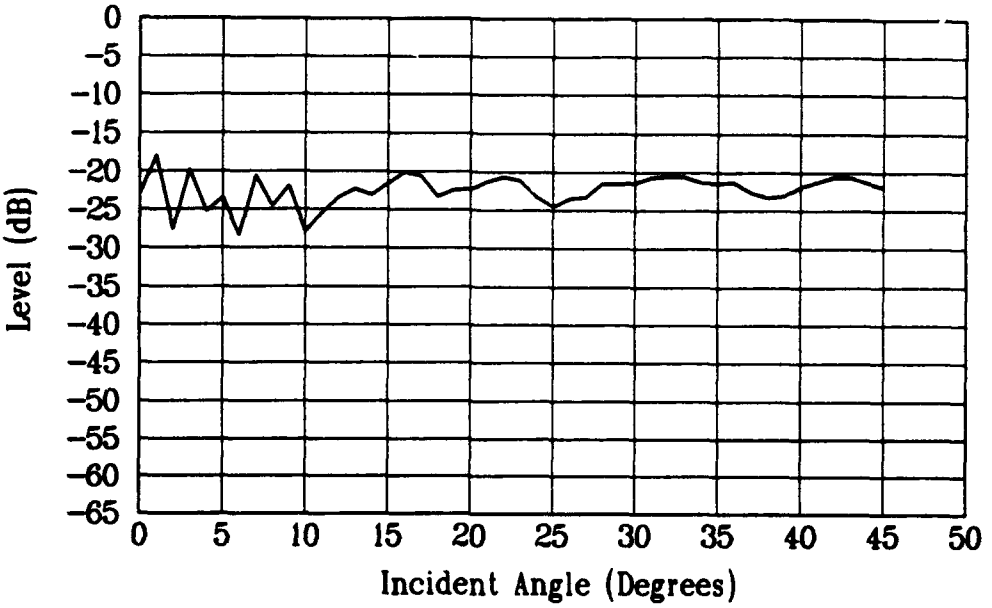

Figure 5.12. Actual levels from 237μm diameter bead surface.

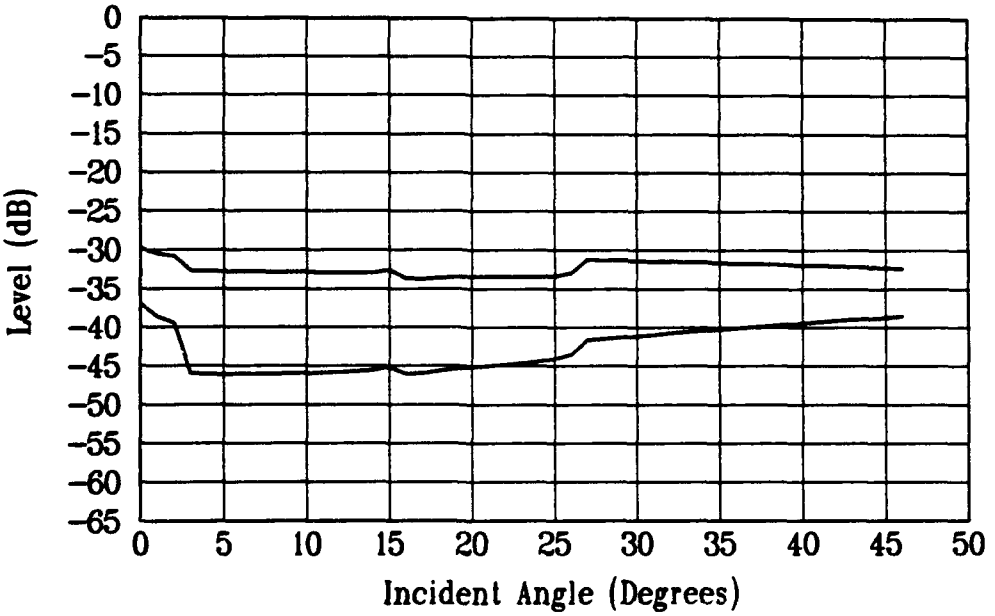

Figure 5.13. Predicted levels from 237μm diameter bead surface.

times the size of R_A. Using similar increases for the other sanded surfaces causes higher than observed levels to be predicted.

For the case of the beaded surface the correlation length is derived from several assumptions. The first of which is a single layer of beads. For this sample the assumption is probably false because of a problem in making the surface. Once this assumption is dropped the value of both R_A and λ_o become unknown. The observed levels for this beaded surface are 10 dB above the higher range expected. This requires an increase of 3.16 times that ratio for a similar result.

Second Experiment Results

The second set of experimental scans has the advantage of providing a visual component as well as intensity levels to evaluate how well ultrasound can characterize roughness. The scans come from two computer systems. The first system is a PDP-11 with a Peritek display. The Peritek display has 512 × 512 pixels and can display 256 levels of gray scale or color. All of the scans except the aluminum cast comparator are made with the PDP-11. This computer however has become inoperable. The final scans are done on a Compaq 386. The loss of the PDP-11 requires a program that can convert the PDP files into IBM compatible PC files. This program, which allows the manipulation of the data for statistical analysis, is documented in the Appendix.

Over fifty scans have been made and of those forty are available for analysis. Of these eighteen will be presented. Table 5.2 lists the scans by file name.

The file names all begin with the letter 'D' and a sequence of numbers that is the date of the scan combined with the incident angle of the scan. The only exceptions are the scans with a 'C' or 'N' after the 'D' which represent the presence of carbon particles or no carbon particles. The scans are listed in Table 5.2.

Table 5.2. Scan data files.

File Name	Size (pixels)	Peak	Level dB	Description
D52115	400 × 400	255	24	Beaded Strips
D52200	400 × 200	255	38	Beaded Strips
D52300	300 × 300	195	46	Resolution Pattern
D52415	300 × 300	255	20	Resolution Pattern
D52600	300 × 200	249	31	168231
D52615	300 × 200	255	31	168231
D52700	160 × 160	226	32	A87065
D52715	160 × 160	244	24	A87065
D52900	500 × 100	255	48	Beaded Strips
D52915	500 × 100	255	27	Beaded Strips
D71900	300 × 200	245	12	088
D71915	300 × 200	226	14	088
DC120	300 × 250	224	30	Carbon Particles
DC1315	300 × 250	252	0	Carbon Particles
DN120	300 × 250	230	30	No Carbon Particles
DN1215	300 × 250	245	0	No Carbon Particles
D70700	512 × 93	213	30	Aluminum Cast Comparator
D70915	512 × 96	203	24	Aluminum Cast Comparator

The following pages present the visual evidence in a 16-step decibel gray scale. The gray scale is illustrated in Figure 5.14. The lowest level, 0, is at the left of the figure. The levels are as represented in Table 5.3. The dynamic range of a 256 level system measuring voltage is 48.1 dB. The colormap has a maximum of 64 levels in a range $0 - 63$. The log display scale shifts the emphasis on higher levels and the variation between those levels.

Figure 5.14. Gray scale.

Table 5.3. Gray scale levels.

Level	Point Range	Palette Level	Peak dB
15	240-255	63	0.0
14	224-239	62	-0.6
13	208-223	60	-1.2
12	192-207	58	-1.8
11	176-191	56	-2.5
10	160-175	54	-3.3
9	144-159	52	-4.1
8	128-143	50	-5.0
7	112-127	47	-6.1
6	96-111	44	-7.2
5	80-95	41	-8.6
4	64-79	37	-10.2
3	48-63	32	-12.1
2	32-47	25	-14.7
1	16-31	16	-18.3
0	0-15	0	-24.6

The scans are presented as pairs, with the 0° scan first, followed by the 15° scan. These are all shown at approximately twice the true scale of the sample. The exception is the aluminum cast comparator scans, which are 1.25 : 1 scale. The scans are presented in roughly chronological order and will be discussed in this order.

The first set of scans is on the resolution block shown in Figures 5.15 and 5.16. The images includes the central 10 millimeter square and some portions of each of the other smaller squares. This result is encouraging, demonstrating that surface roughness differences can be observed. This also demonstrates that for extremes in roughness even the normal scan can show differences in surface roughness.

The next two sets of scans are of the same surface, the beaded strips. The first set, D52115 and D52200, are of the same area however the 0° scan is only half the size because of the time required to make the larger scan and poor scans that resulted due to computer, stepper motor, and transducer problems with long scan times (longer than 24 hours). The larger scan does overlap the region covered in the small scan. Several features can be seen. The outer two rows of larger beads are not in these scans. In the 0 degree scan the larger two sizes of beads in the center scatter more of the energy away from the specular direction than the three rows of smaller beads on either side. In the 15° scan only one of the larger bead rows appears to make a larger contribution to the reflected angle energy.

Figure 5.15. Resolution block 0 degrees.

Figure 5.16. Resolution block 15 degrees.

Figure 5.17. D52200; bead strips 0 degrees.

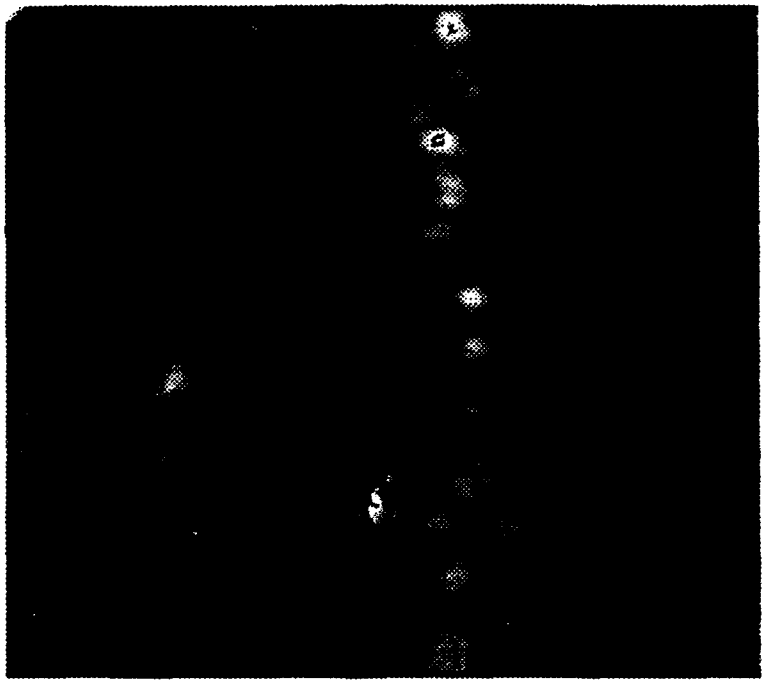

Figure 5.18. D52115; bead strips 15 degrees.

The second set of bead strips does include portions of the ten strips. The scans are shown in Figures 5.19 and 5.20. Again the number of scan lines was reduced for time considerations. In Figure 5.19 the larger beads in the center and at the edges have a lower return than the other beads in the 0° scan. The reverse is true for the 15° scan.

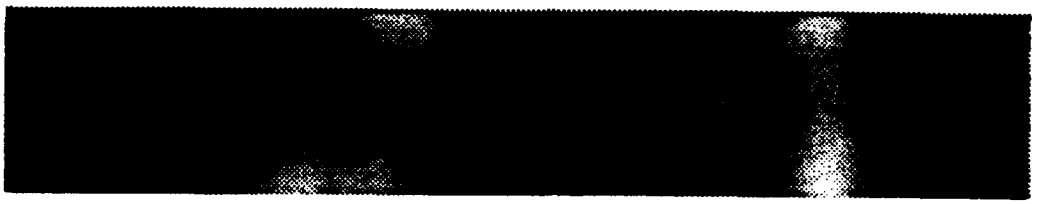

Figure 5.19. D52900; bead strips 0 degrees.

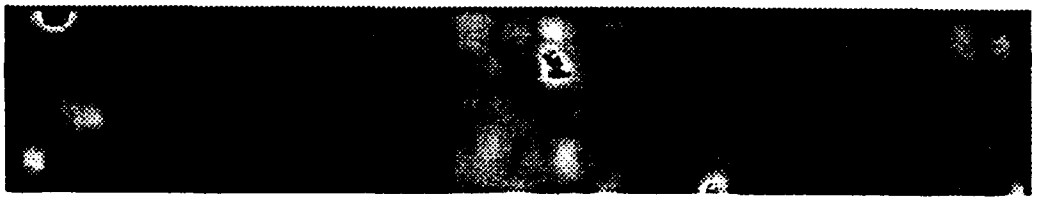

Figure 5.20. D52915; bead strips 15 degrees.

The next two sets of scans are of plexiglas blocks with regions that have glue applied. In the first surface, two areas of carbon particles sprinkled on glue are separated by a sanded region (280 grit). The sanded region is a central strip in the middle of the block from top to bottom and across the top of the right hand side. The area to the left side has particles of $10\mu m$. The area to the right has particles of $60\mu m$.

The second surface has only the glue on the two areas separated by a sanded region. The carbon particle scans are shown in Figures 5.21 and 5.22.

The glue only scans are shown in Figures 5.23 and 5.24. There are some interesting regions that can be seen in these figures. In both of the 0° scans the region of sanded surface appears uniform. The particles on glue region and the glue only region have a similar undulating appearance. The 15° scans also look quite similar. Unfortunately the regions of highest return do not seem to be aligned with either of the regions of particles on glue or glue only. Another feature that can be seen is many small dark dots. These are pixels with very low levels (< 12). They appear to be data drop out problems with the digitizer board on the PDP-11. They appear uniformly throughout all four scans. The worst case is in Figure 5.22, which is also the last of the four to be made.

The next set of scans is of the aluminum cast comparator. The scans are shown in Figures 5.25 and 5.26. The roughest surface is on the left. The highest levels on the 0° scan appear to be from the central region. This is not the expected result. Upon further investigation the plate was found to have a slight curvature to the surface. Because of the sensitivity of the scattering function around 0° the slight curvature is enough to cause a disruption in the expected result. Measurements of the plate were made to determine the curvature. The result is that over the whole plate the normal to the surface varied by 1° to 2° depending on where the measurements are taken. The 15° scan shows somewhat similar problem with the highest levels coming from the roughest areas (expected) and from the central region again (unexpected). Because of the curvature the angle of incidence is varying around the region of the critical angle for aluminum of 13.6°. Nonetheless the scan does show variations for each surface roughness area.

Figure 5.21. DC120; carbon particles 0 degrees.

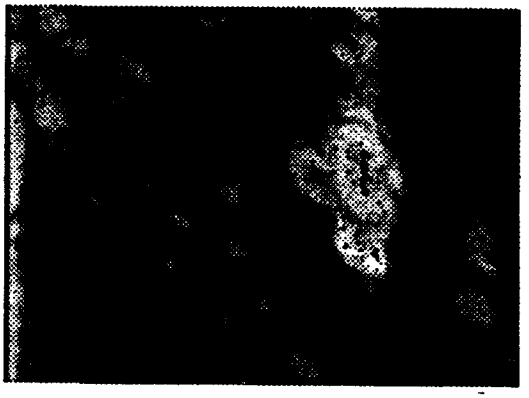

Figure 5.22. DC1315; carbon particles 15 degrees.

The next three sets of scans are of tissue samples. As noted before in Chapter 3, the surfaces retained some of their natural curvature when open and flat. The scans here are demonstrations of the feasibility. They show that as the angle is changed some variation will occur and the cause of the changed level is the variation in the surface. Unfortunately most of the variation seen in the scans is due to the undulation from a curved cylindrical surface reacting to internal stresses rather than the actual roughness of the surface in its normal

Figure 5.23. DN120; no carbon particles 0 degrees.

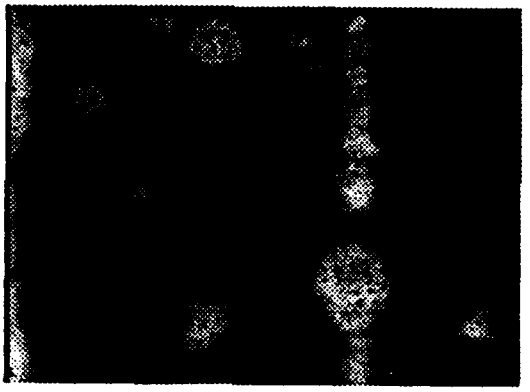

Figure 5.24. DN1215; no carbon particles 15 degrees.

state. Figures 5.27 and 5.28 show a relatively normal sample. This is the same sample shown in Figure 3.10. The regions of highest return are from the curve of the surface. Figures 5.29 and 5.30 show another relatively normal sample. There are regions here that have a large number of data dropouts in regions of higher return. Again the regions of highest return are from the curve of the surface. Figures 5.31 and 5.32 are the scans that correspond to the highly calcified tissue pictured in Figure 3.11. Here there is some indication by more

Figure 5.25. D70700; aluminum cast comparator 0 degrees.

Figure 5.26. D70915; aluminum cast comparator 15 degrees.

frequent undulation of levels that the plaque is being shown as well as the tendency of the surface to recurve itself back into a cylinder.

The next section describes some statistical relationships found by using the program documented in the Appendix. The regions of various scans selected are shown again with the areas of interest drawn as rectangles on the scans. These levels are compared with the plots of expected scattering strength as in the first experiment set. Because the source of disagreement in the first experiment set appears to be correlation length, an attempt is made to find the correlation length from the backscatter field.

For each of the scans the following analysis is made. A section of a scan is selected that has the same value of roughness. The data points are averaged and the standard deviation found for that area. The decibel level is computed for the average value, the average value plus one standard deviation and the average value minus one standard deviation. The decibel level is referenced

Figure 5.27. D52700; A87065 tissue sample 0 degrees.

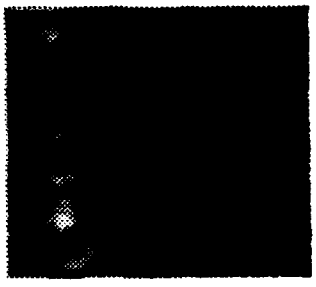

Figure 5.28. D52715; A87065 tissue sample 15 degrees.

to the absolute peak level in the 0° scan. The assumption is that the highest level is equivalent to a plane surface somewhere in the scan. Because there is a difference in attenuation for the 15° levels, an adjustment is made by subtracting the difference,

$$I_{15} = I_{15,measured} - (A_0 - A_{15}) \qquad 5.1$$

where A_0 is the attenuation level for the 0° scan and A_{15} is the attenuation for the 15° scan. In addition, for each rectangular area selected, at least five lines were chosen to find the correlation lengths for the acoustic field. Finally the three averaged levels for 0° and 15° are compared with a plot of the the predicted backscatter levels using the calculated or derived values of

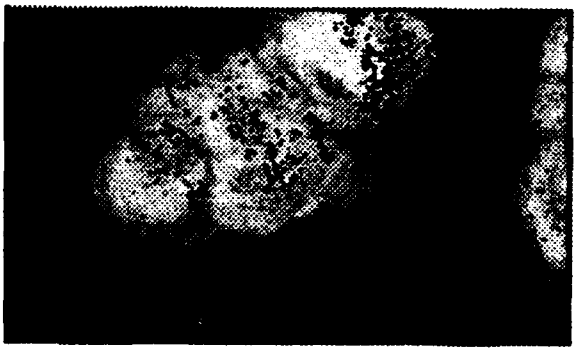

Figure 5.29. D71900; 088 tissue sample 0 degrees.

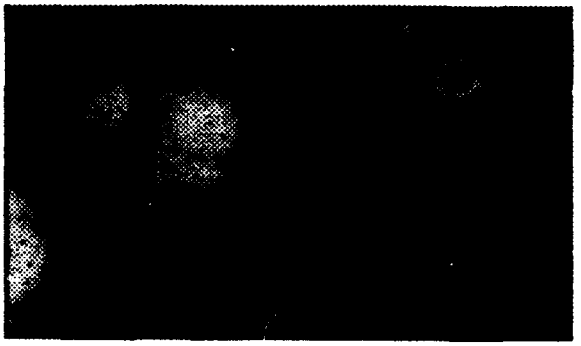

Figure 5.30. D71915; 088 tissue sample 15 degrees.

the correlation length set in the first experiment set and with predicted levels using the average correlation length from the acoustic field.

The first surface scan to be analyzed is the resolution square. Figures 5.33 and 5.34 show the scans and the two regions used to compute the averages. The regions are enclosed in rectangles. The average level and the standard deviation of the levels in each region as well as the correlation lengths of lines in each region are calculated by the computer program. The average levels and the average correlation length are shown in Table 5.4. The roughness of

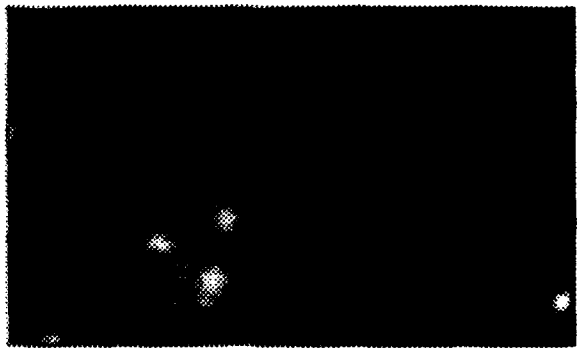

Figure 5.31. D52600; 168231 tissue sample 0 degrees.

Figure 5.32. D52615; 168231 tissue sample 15 degrees.

the rough area is unknown. The smooth area however was made with 400 grit sandpaper. Because the sanded surface has a value g much less than unity, equations 2.49 and 2.50 are used for the prediction curves. Figure 5.34 shows how the levels compare with the correlation values from Table 3.1. In this figure, as in all the following the average level measured is marked with a circle on the graph, o. If more than one area is measured for the same surface roughness, there is a circle for each average. In addition the plus and minus one standard deviation levels are marked with a diamond, ◇. Thus the

extent of the markings at 0 and 15 degrees show the confidence interval for the measured values. The average correlation length for the acoustic field was $1340\mu m$. The comparison is shown in Figure 5.35.

Table 5.4. Backscatter levels for 400 sanded surface.

Angle	Average − S.D.	Average Level	Average + S.D.
0	-4.8	-4.3	-3.8
15	-51.6	-45.3	-41.7

The other scans that include a portion of sanded surface are the four with and without carbon particles. The correlation length for the sanded regions was calculated from the two central regions. The scan areas are shown in Figures 5.37 and 5.38. Only the scans with the carbon particles are shown because there is essentially no difference between the carbon and carbonless scans. Figure 5.39 shows the average backscatter levels compared with the predicted values using λ_o from the Table 3.1. The average levels and the standard deviations are shown in Table 5.5. The sanded area was divided into three segments on the two types of scans. Therefore there are a total of six entries for each angle. The values are presented in the following order. The first three values for each angle are from the scan with the carbon particles. The next three are from the scan without the carbon particles. Of the three, the first is from the upper central rectangle. The second of the three is from the lower central rectangle. The third of three is from the upper right rectangle. The values are reasonably close to one another. Figure 5.40 shows the comparison of the observed backscatter levels compared with the predicted level with the acoustic field value for the correlation length of $1385\mu m$.

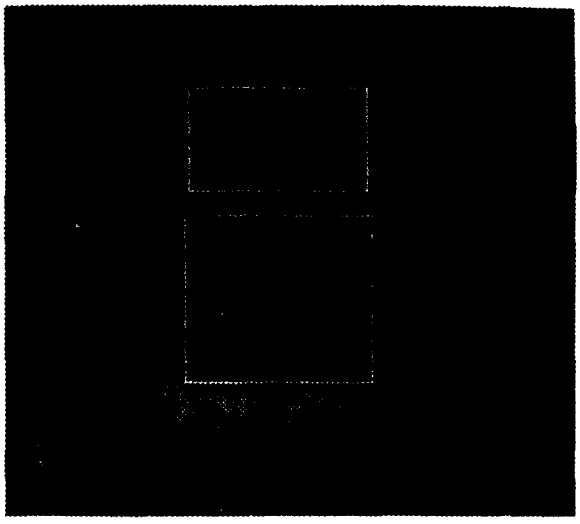

Figure 5.33. Resolution block 0 degrees.

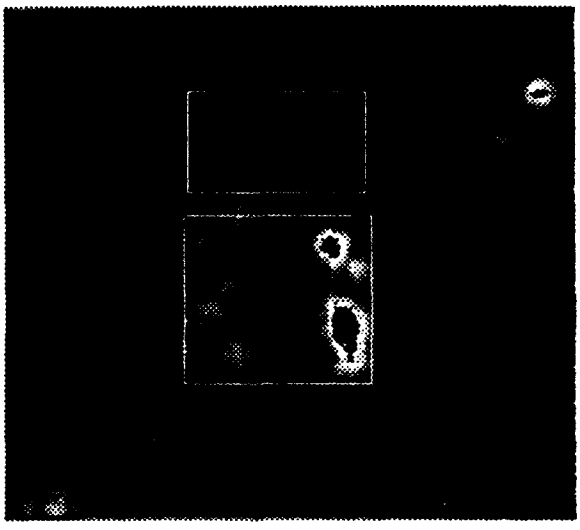

Figure 5.34. Resolution block 15 degrees.

The results for the beaded strips are presented in Table 5.6 in which data are presented for each angle a maximum of four times. There are two strips on the surface and because there are two sets of scans; D52200 and D52115

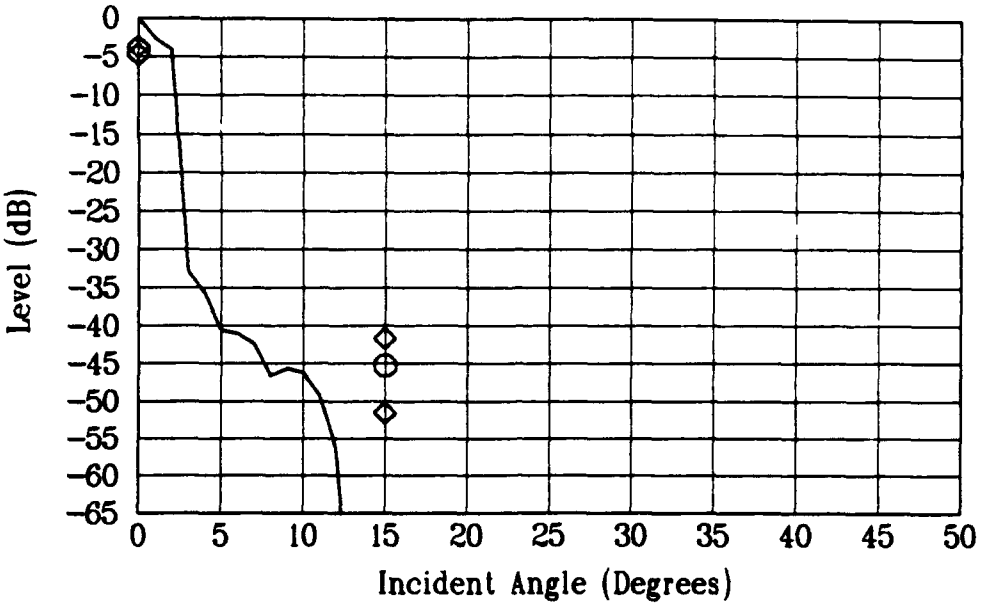

Figure 5.35. Resolution block with material λ_o.

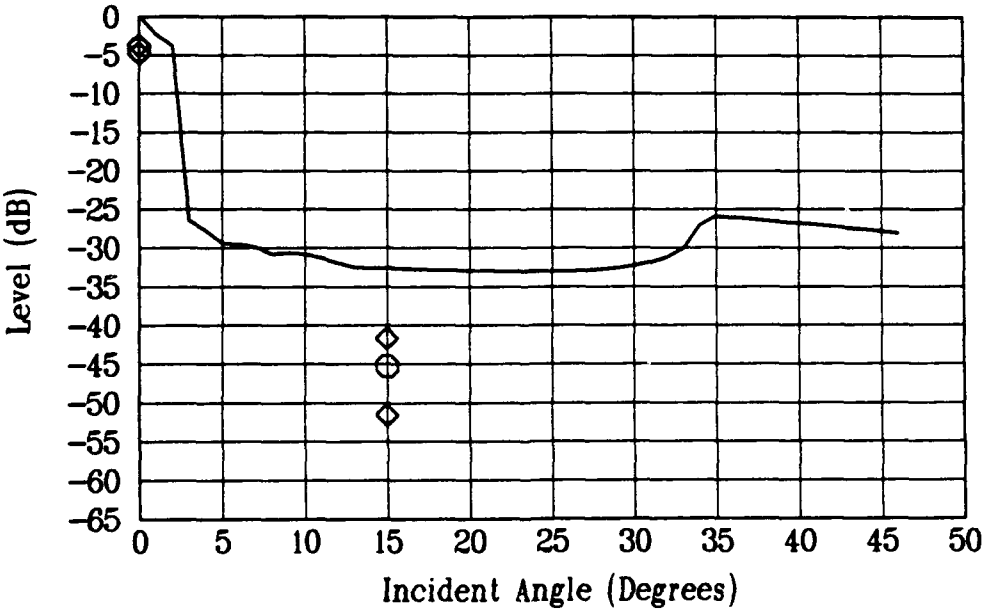

Figure 5.36. Resolution block with acoustic λ_o.

Table 5.5. Backscatter levels for 280 sanded surface.

Angle	Average − S.D.	Average Level	Average + S.D.
0	-3.7	-3.2	-2.6
	-3.7	-3.0	-2.4
	-3.1	-2.5	-1.8
	-5.5	-4.6	-3.7
	-5.1	-4.4	-3.7
	-4.6	-3.6	-2.8
15	-42.6	-38.5	-35.8
	-39.8	-35.3	-32.4
	-39.9	-36.8	-34.6
	-41.5	-38.0	-37.9
	-46.1	-37.8	-33.6
	-43.4	-39.3	-36.6

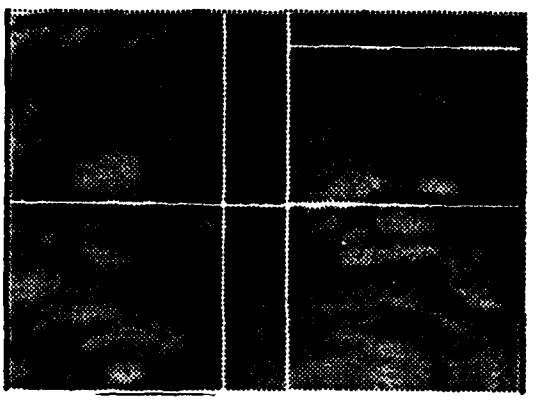

Figure 5.37. DC120; carbon particles 0 degrees.

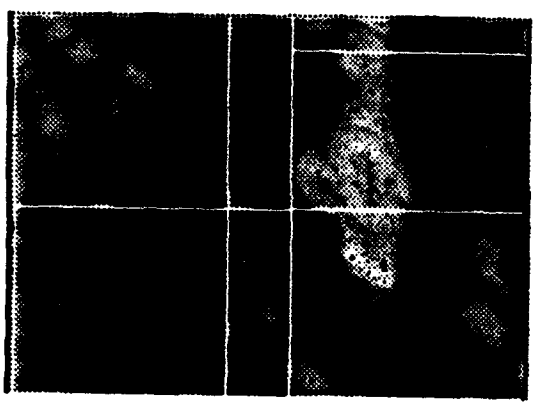

Figure 5.38. DC1315; carbon particles 15 degrees.

as well as D52900 and D52915. The values vary greatly at each angle. The first set has only eight columns as can be seen in Figures 5.41 and 5.42. The second set has ten areas as shown in Figures 5.43 and 5.44.

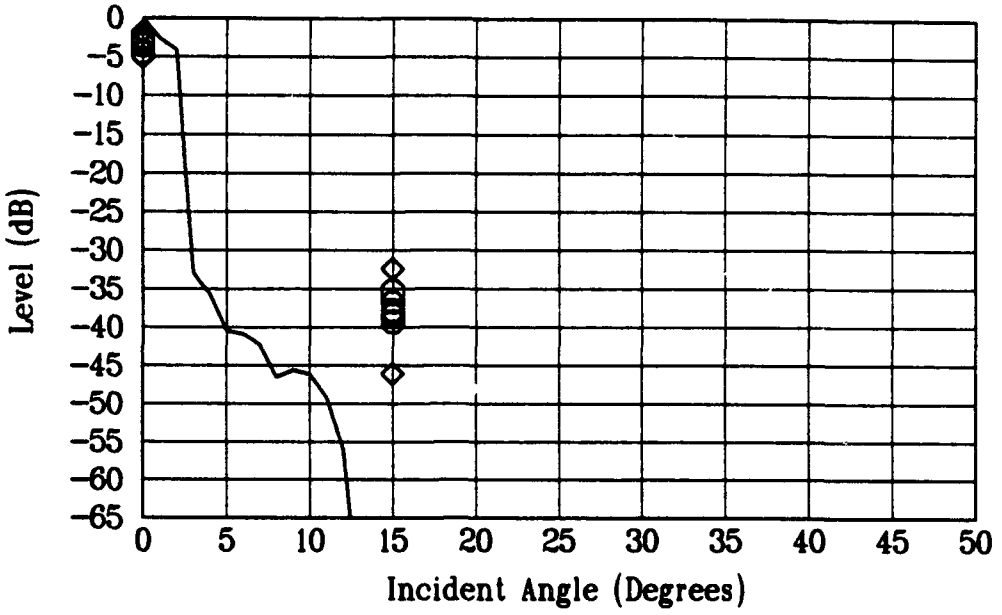

Figure 5.39. 280 Grit Sanded With Material λ_o.

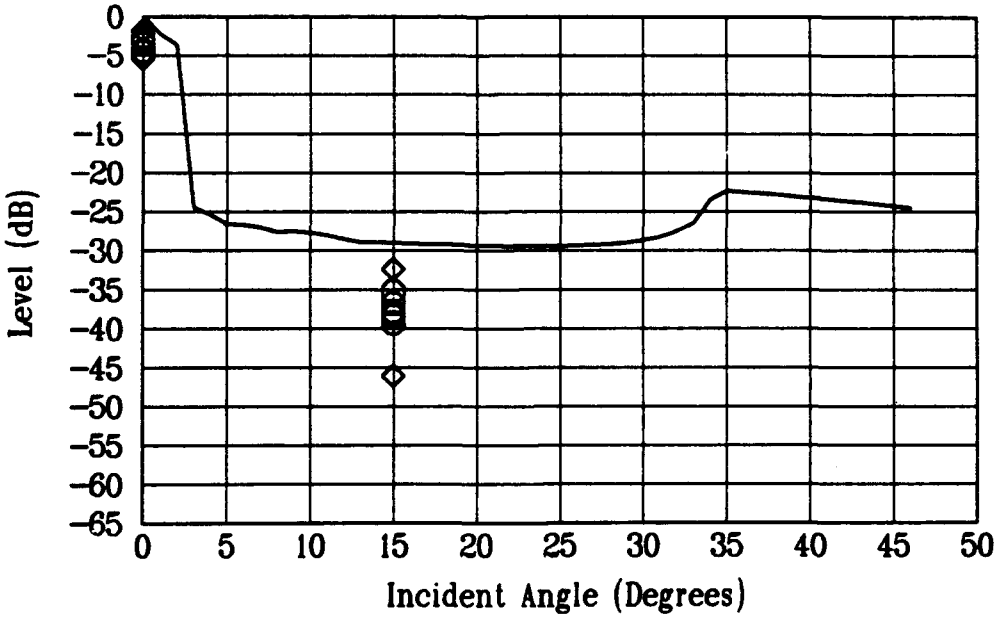

Figure 5.40. 280 Grit Sanded With Acoustic λ_o.

Table 5.6. Backscatter levels for beaded strips.

Bead Size	Angle	Average − S.D.	Average Level	Average + S.D.
50	0	-10.4	-6.8	-4.3
		-16.2	-10.4	-6.9
		-15.7	-11.0	-7.9
		-18.4	-10.4	-6.3
50	15	-35.0	-31.2	-28.6
		-34.4	-31.6	-29.5
		-29.6	-24.8	-21.8
		-28.4	-25.7	-23.6
105	0	-13.7	-8.2	-4.9
		-8.5	-4.7	-2.0
		-17.1	-9.1	-5.0
		-14.1	-8.9	-5.7
105	15	-34.4	-31.3	-29.0
		-32.4	-29.7	-27.7
		-29.5	-25.1	-22.3
		-28.4	-25.2	-22.9
150	0	-15.9	-9.5	-5.9
		-15.1	-9.3	-5.8
		-16.1	-12.6	-10.1
		-16.0	-8.7	-4.8
150	15	-35.9	-32.2	-29.7
		-32.7	-29.1	-26.6
		-27.2	-24.6	-22.6
		-29.6	-23.1	-19.4
215	0	-22.7	-14.0	-9.7
		-37.5	-19.4	-14.0
		-23.5	-16.7	-12.9
215	15	-31.5	-27.4	-24.6
		-30.4	-26.5	-23.7
		-27.2	-22.8	-19.9
237	0	-32.4	-17.6	-12.4
		-29.9	-20.0	-15.5
		-27.0	-21.5	-18.1
237	15	-30.1	-26.6	-24.1
		-30.7	-28.0	-25.9
		-28.6	-25.7	-23.6

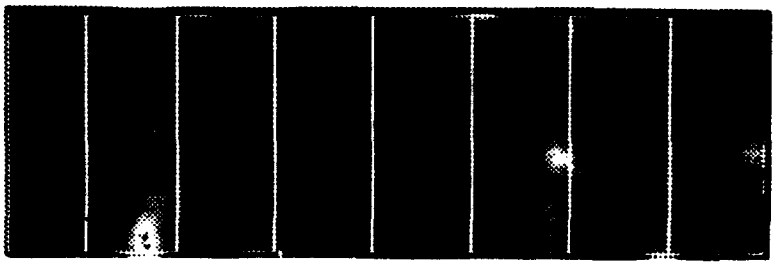

Figure 5.41. D52200; bead strips 0 degrees.

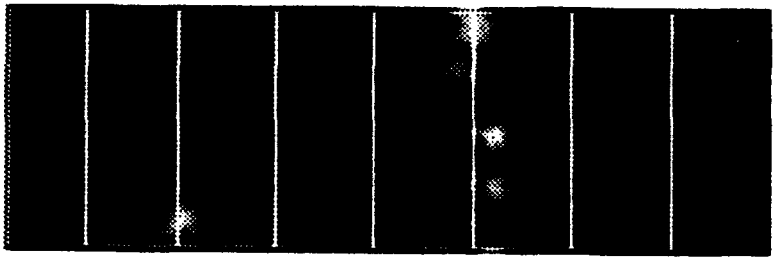

Figure 5.42. D52115; bead strips 15 degrees.

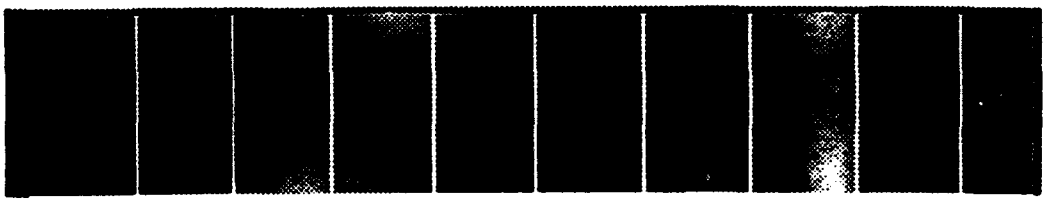

Figure 5.43. D52900; bead strips 0 degrees.

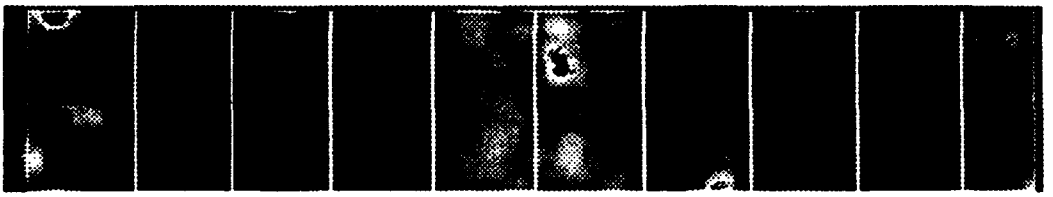

Figure 5.44. D52915; bead strips 15 degrees.

The predicted levels for backscatter level from the beaded surfaces use equations 2.49 and 2.51 or equation 2.52 alone, depending on the value of g. For the two smallest sizes of beads equations 2.49 and 2.51 are used. For the other sizes both equations 2.49 and 2.51 as well as 2.52 are used and compared. This is done because the value of g for these sizes is on the borderline for the conditions governing the theory. If equations 2.49 and 2.51 are used, there are two lines plotted that establish the predicted range of backscatter intensity. If equation 2.52 is used there is only a single line shown. Figures 5.45 and 5.46 show the backscatter levels plotted on the predicted levels for the $50\mu m$ beaded surface for the calculated λ_o from Table 3.2 and for the measured acoustic λ_o, respectively. Figures 5.47 through 5.50 show similar plots for the $105\mu m$ and $150\mu m$ diameter bead surfaces using equations 2.49 and 2.51. Figures 5.51 and 5.52 show plots for the $150\mu m$ diameter bead surface using equation 2.52 assuming that g is much greater than 1. The next four plots show the measured data for the $215\mu m$ diameter bead surface. Figures 5.53 and 5.54 show the observed data compared to equations 2.49 and 2.51. Figures 5.55 and 5.56 show the observed data again using equation 2.52. Figures 5.57 through 5.60 show similar plots for the $237\mu m$ diameter bead surface using equations 2.49 and 2.51, and 2.52.

The last surface to be shown is the aluminum cast comparator. The problem with the curvature has already been discussed. The backscatter strength levels are presented in Table 5.7. The spread of levels across the one standard deviation is much tighter than for the beads. The trend expected of higher levels at 0° incidence angle for the least rough is not seen. The reverse trend for the 15° incidence angle is also not consistently present. The regions used to make the averages are shown in Figures 5.61 and 5.62. Comparisons

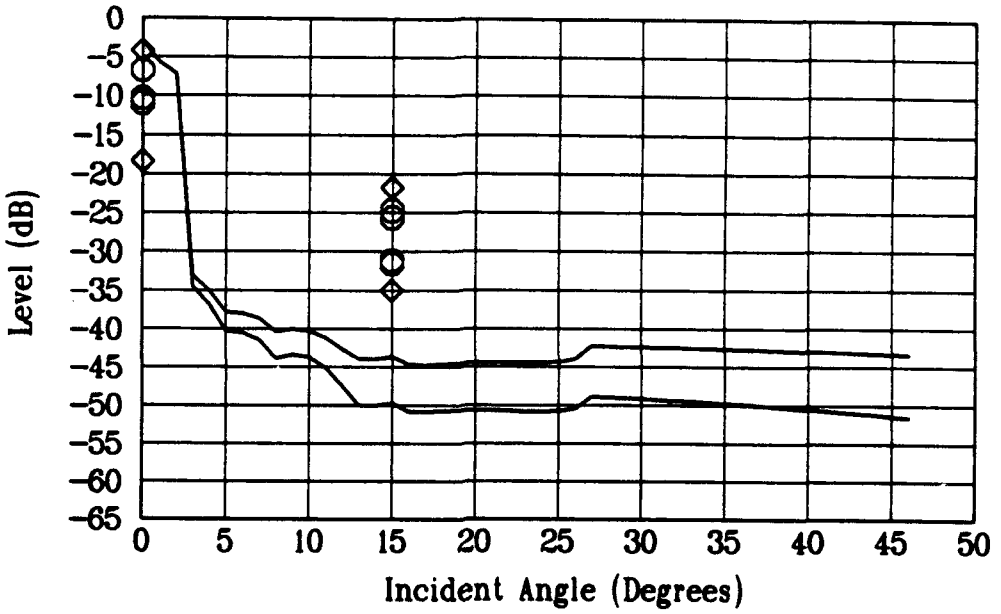

Figure 5.45. $50\mu m$ bead surface ($g \sim 1$) with material λ_o.

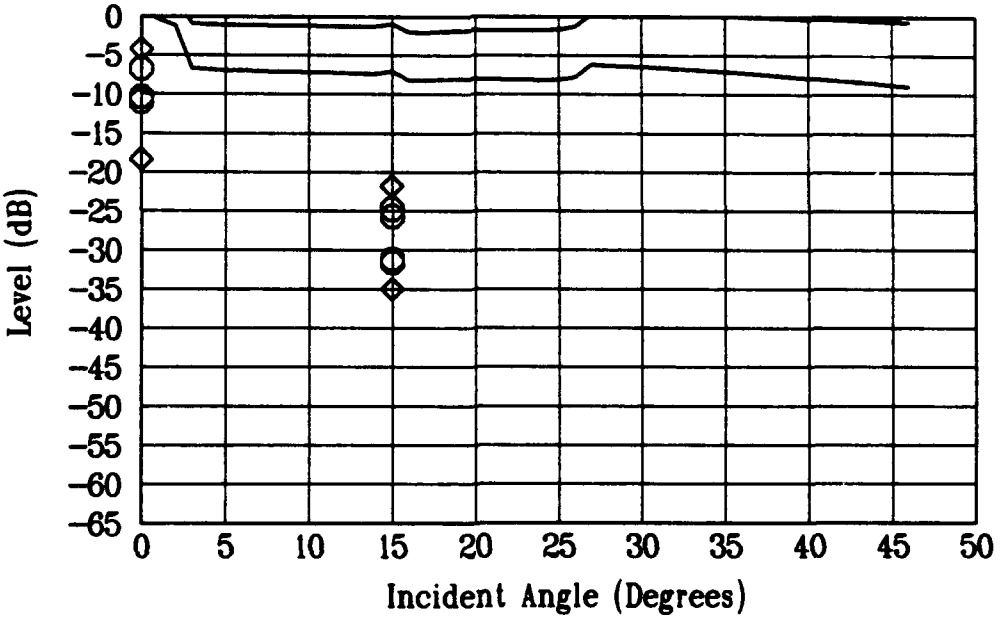

Figure 5.46. $50\mu m$ bead surface ($g \sim 1$) with acoustic λ_o.

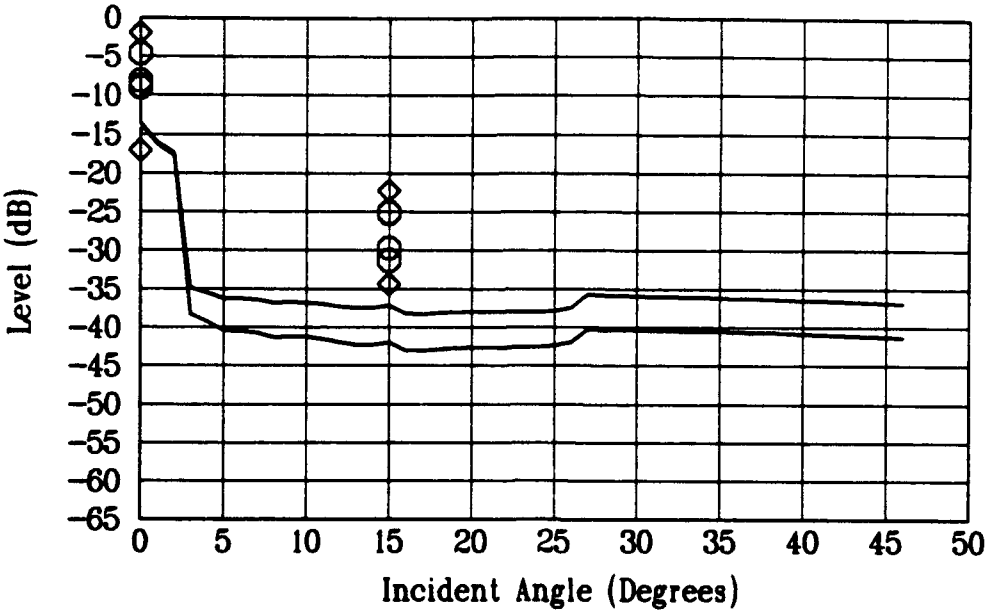

Figure 5.47. $105\mu m$ bead surface ($g \sim 1$) with material λ_o.

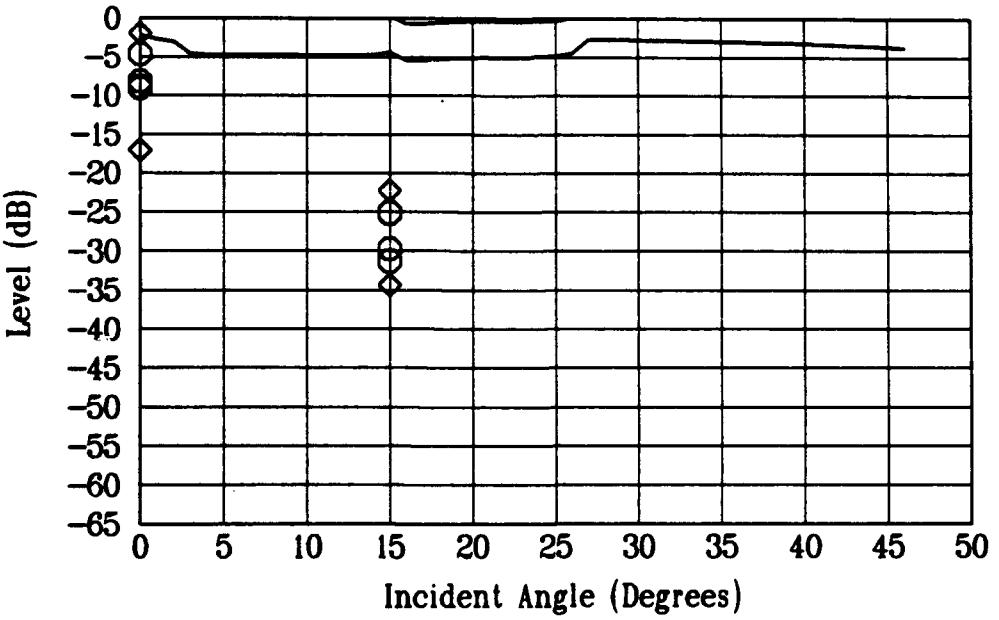

Figure 5.48. $105\mu m$ bead surface ($g \sim 1$) with acoustic λ_o.

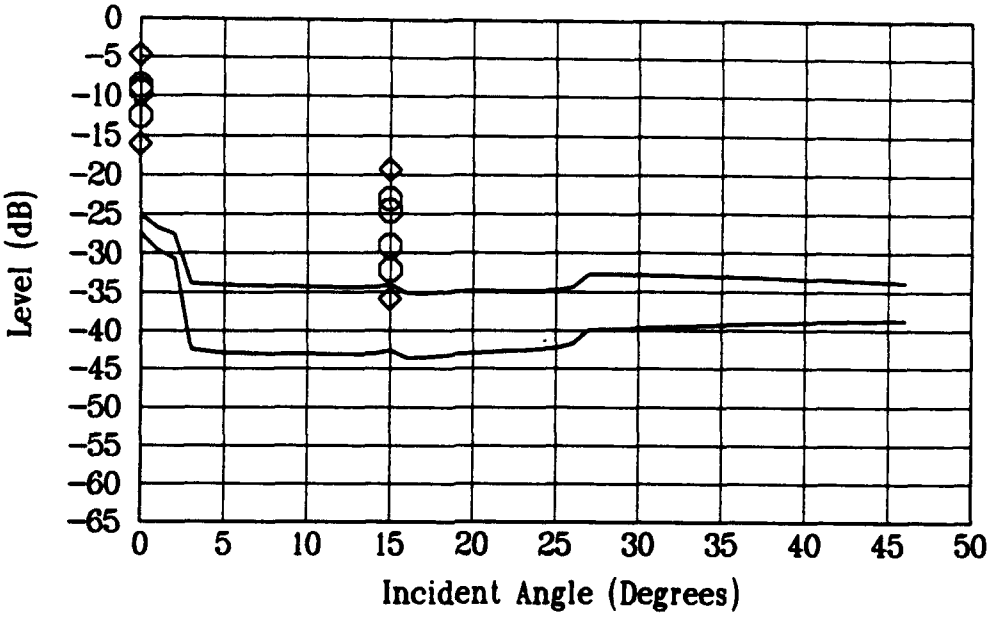

Figure 5.49. $150\mu m$ bead surface ($g \sim 1$) with material λ_o.

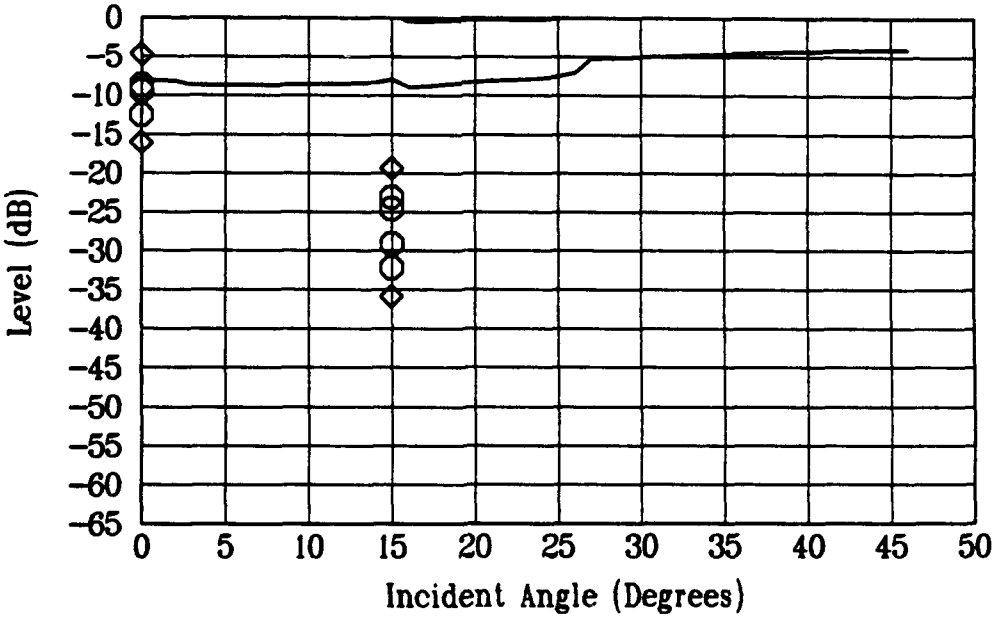

Figure 5.50. $150\mu m$ bead surface ($g \sim 1$) with acoustic λ_o.

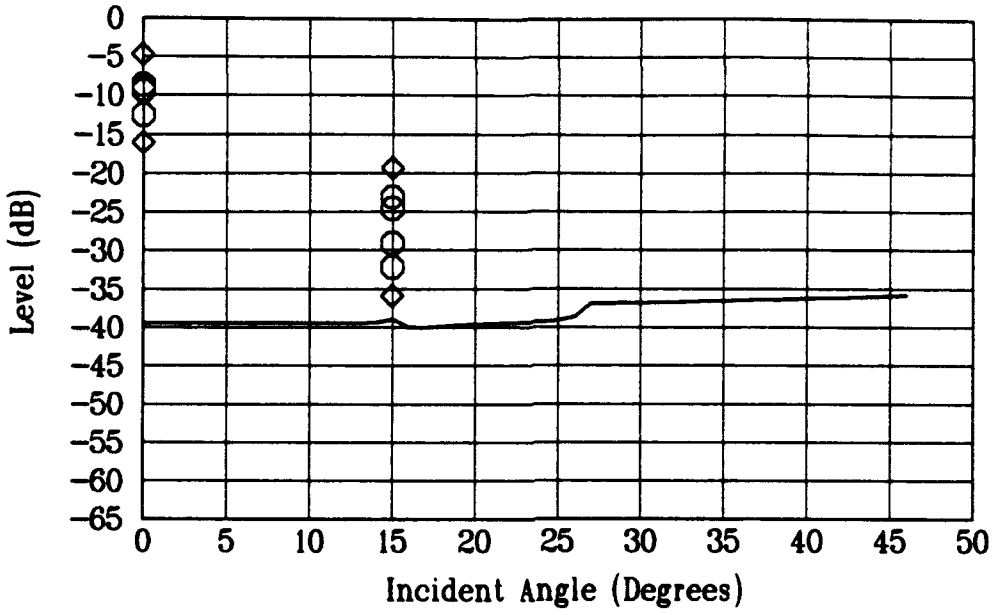

Figure 5.51. $150\mu m$ bead surface ($g \gg 1$) with material λ_o.

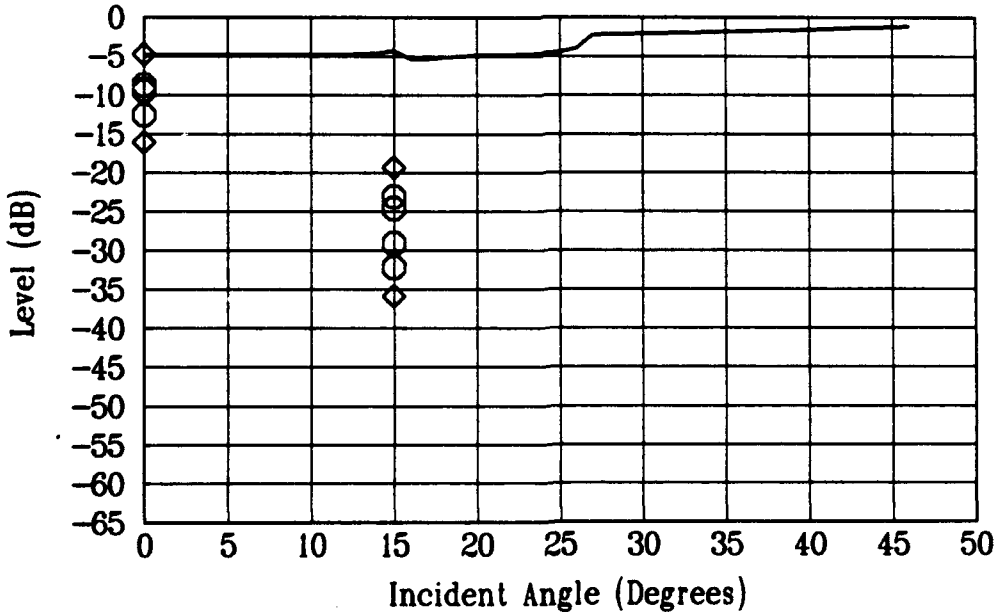

Figure 5.52. $150\mu m$ bead surface ($g \gg 1$) with acoustic λ_o.

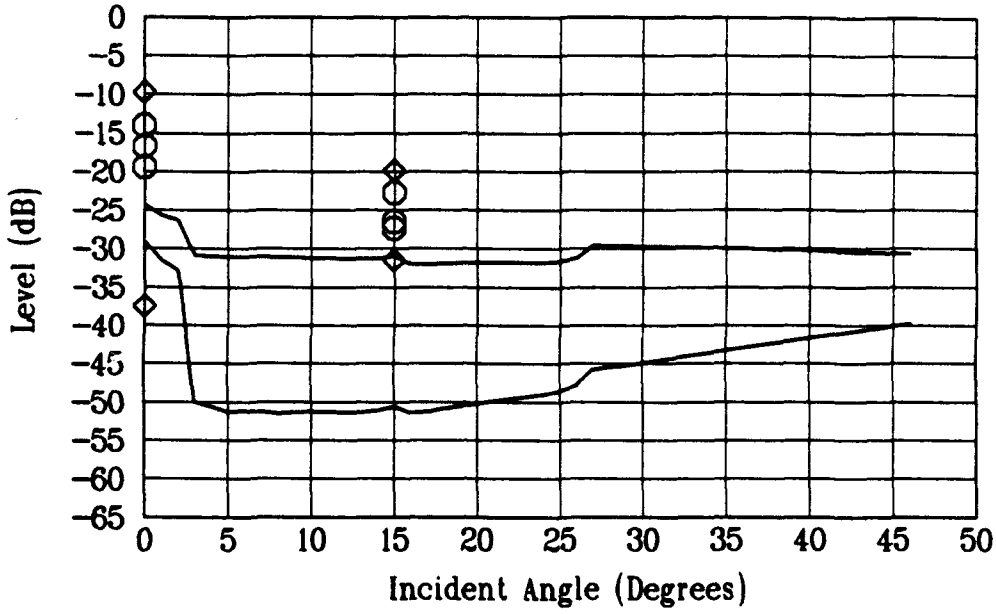

Figure 5.53. $215\mu m$ bead surface ($g \sim 1$) with material λ_o.

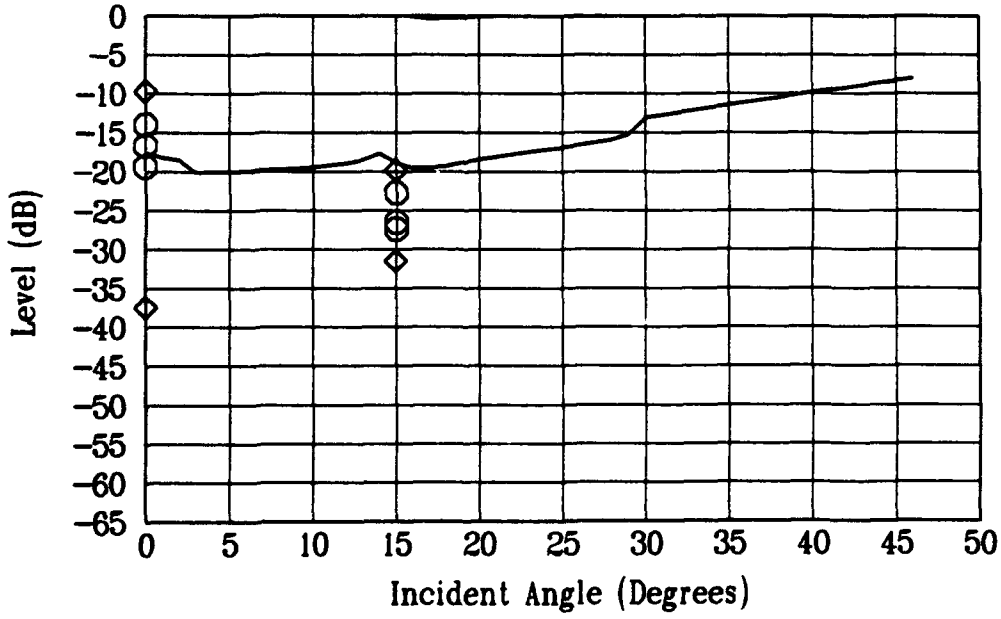

Figure 5.54. $215\mu m$ bead surface ($g \sim 1$) with acoustic λ_o.

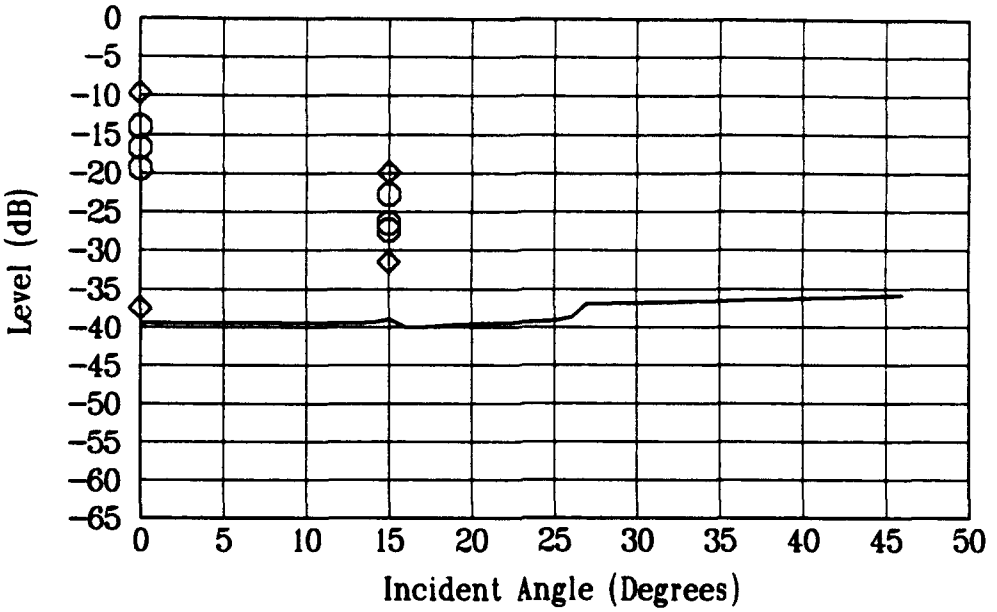

Figure 5.55. 215μm bead surface ($g \gg 1$) with material λ_o.

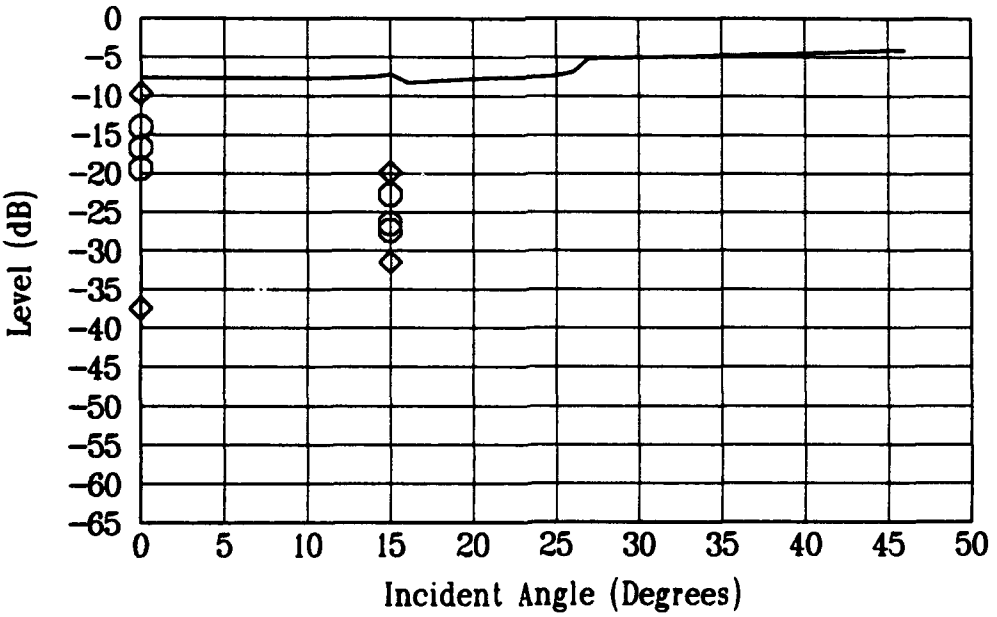

Figure 5.56. 215μm bead surface ($g \gg 1$) with acoustic λ_o.

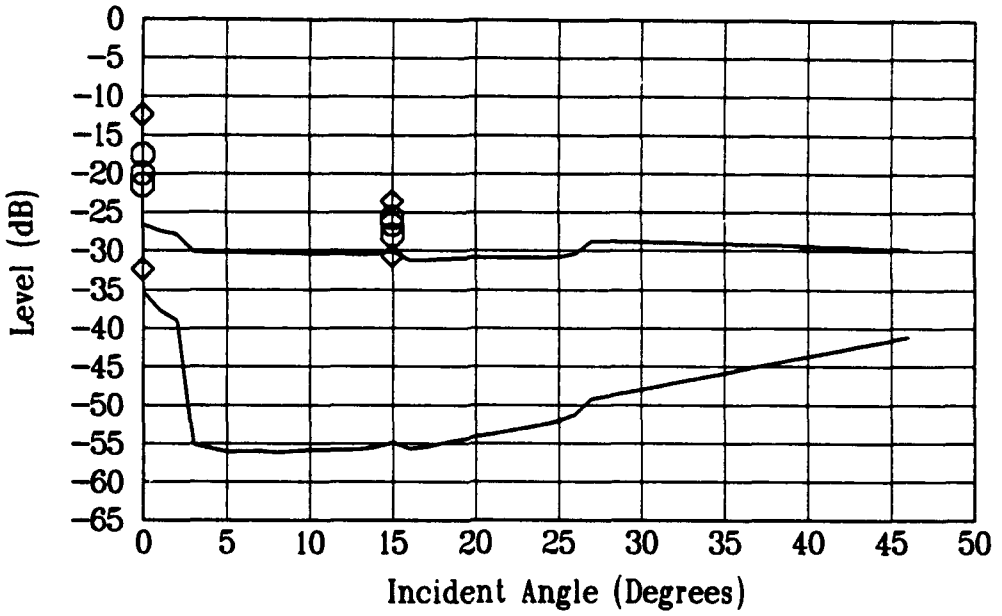

Figure 5.57. 237μm bead surface ($g \sim 1$) with material λ_o.

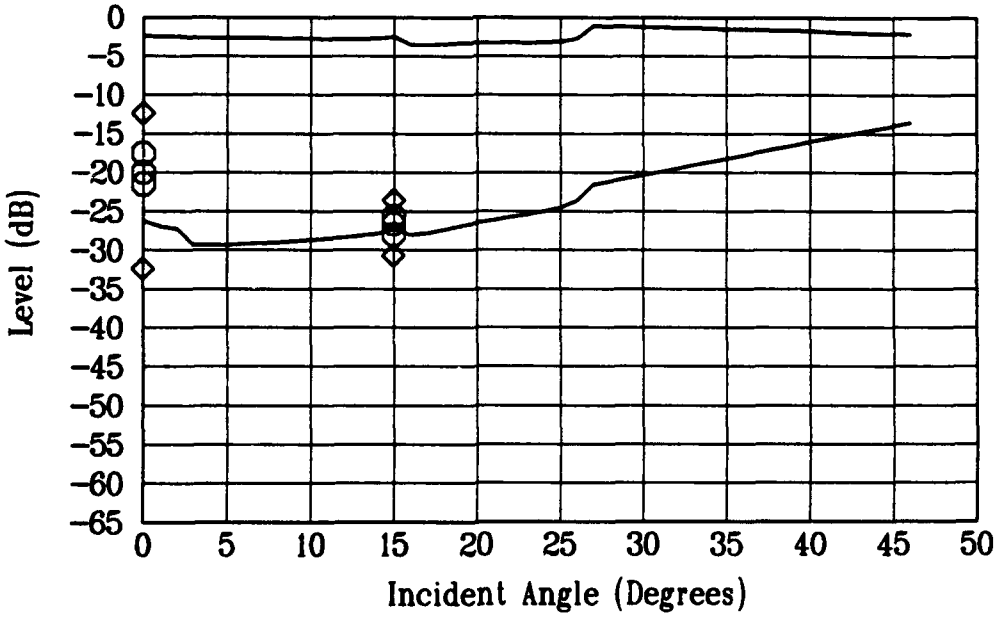

Figure 5.58. 237μm bead surface ($g \sim 1$) with acoustic λ_o.

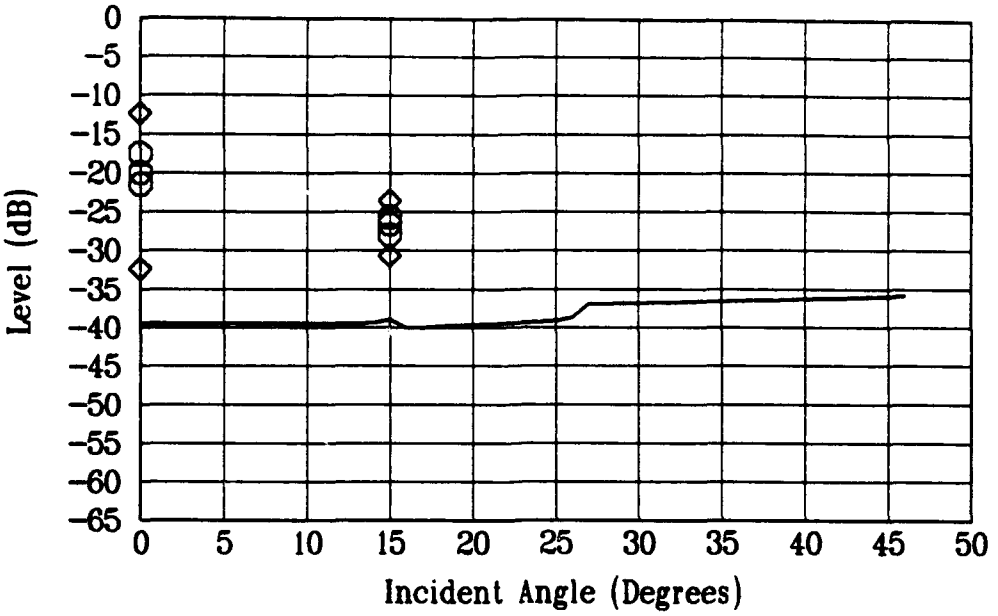

Figure 5.59. 237μm bead surface ($g \gg 1$) with material λ_o.

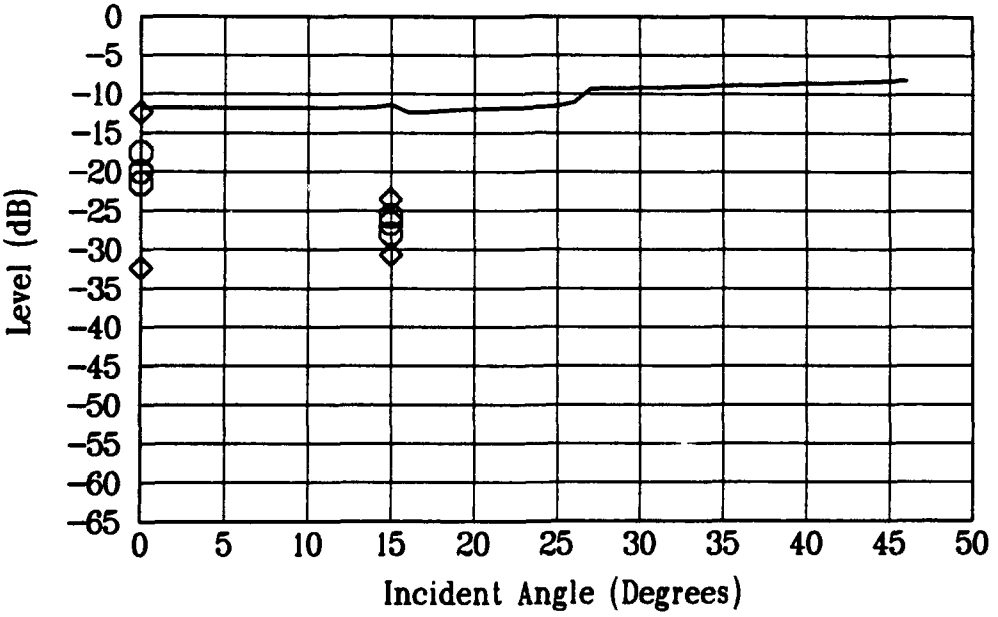

Figure 5.60. 237μm bead surface ($g \gg 1$) with acoustic λ_o.

of the correlation length values estimated from Table 3.3 and the measured values using the acoustic field are shown in Figures 5.63 through 5.84. The first three surfaces have roughness values that meet the following condition: $g \ll 1$. These are illustrated in Figures 5.63 through 5.68. The next four surfaces meet the condition, $g \sim 1$. The comparisons for these surfaces are shown in Figures 5.69 through 5.76. The two curves illustrate the upper bound and the lower bound. For the last two surfaces the condition, $g \sim 1$, or the condition, $g \ll 1$, might apply so both are shown. These are in Figures 5.75 through 5.84. For some of the acoustic field correlation lengths the predicted field has a gain predicted that is impossible. This casts considerable doubt on the relationship between the measured acoustic field correlation length and its actual value. Table 5.8 compiles a list of all the roughness of the surfaces in this study along with the correlation lengths, both found or derived and the acoustic field correlation length. In addition the table gives the values of g for all the materials. Tables 5.9 and 5.10 compile the errors for all the surfaces at 0° and at 15° , respectively. The levels for each surface type take the average of the observed levels if more than one value exists for a particular surface. For the conditions where equation 2.51 is used, the error is only listed if it is outside the bounds of the two curves. The error is calculated from the nearest line.

Table 5.7. Backscatter levels for cast comparator.

R_A μm	Angle	Average − S.D.	Average Level	Average + S.D.
0.5	0	-13.2	-11.5	-10.1
	15	-33.2	-29.5	-26.9
1.5	0	-11.6	-9.1	-7.1
	15	-32.6	-29.9	-27.9
3.0	0	-11.2	-9.1	-7.4
	15	-28.0	-25.3	-23.2
5.1	0	-9.7	-7.4	-5.6
	15	-24.3	-21.9	-20.0
7.6	0	-5.3	-3.4	-1.8
	15	-22.3	-19.8	-17.9
10.7	0	-4.8	-2.9	-1.4
	15	-24.4	-21.8	-19.8
14.2	0	-8.6	-5.8	-3.7
	15	-24.4	-21.4	-19.1
18.3	0	-12.5	-8.7	-6.0
	15	-21.3	-17.6	-15.0
22.9	0	-13.5	-10.4	-8.1
	15	-22.1	-19.0	-16.7

Figure 5.61. D70700; aluminum cast comparator 0 degrees.

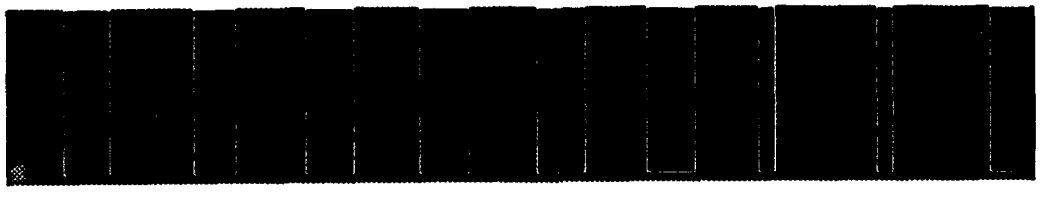

Figure 5.62. D70915; aluminum cast comparator 15 degrees.

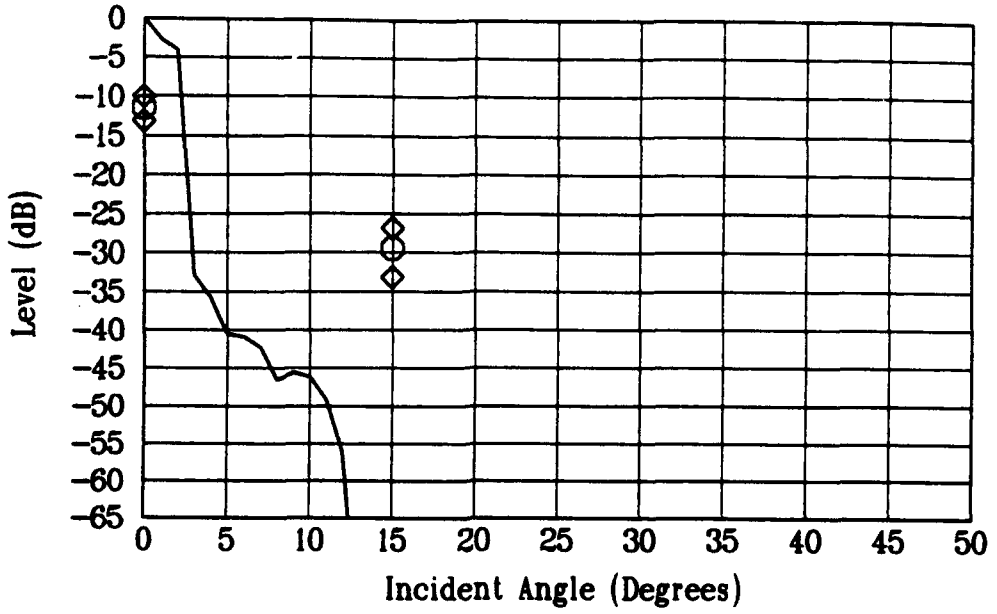

Figure 5.63. $0.51\mu m R_A$ aluminum surface $g \ll 1$ with material λ_o.

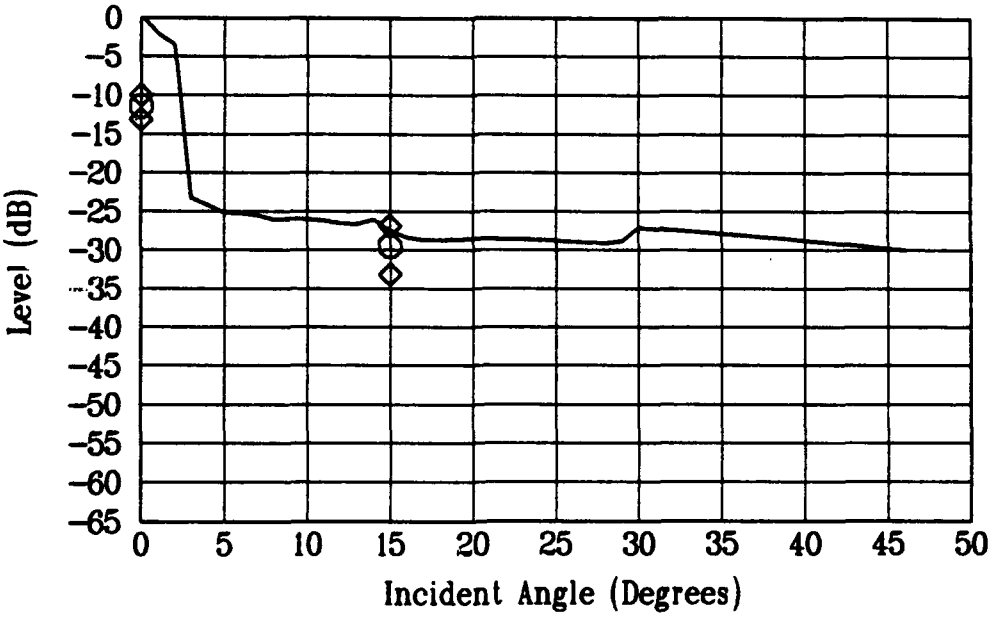

Figure 5.64. $0.51\mu m R_A$ aluminum surface $g \ll 1$ with acoustic λ_o.

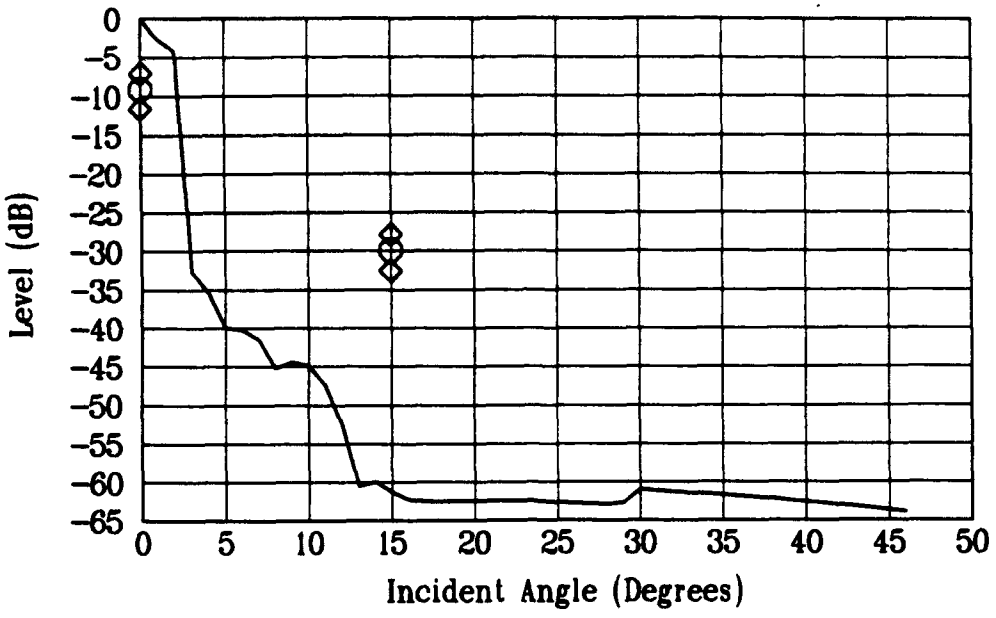

Figure 5.65. $1.53\mu m R_A$ aluminum surface $g \ll 1$ with material λ_o.

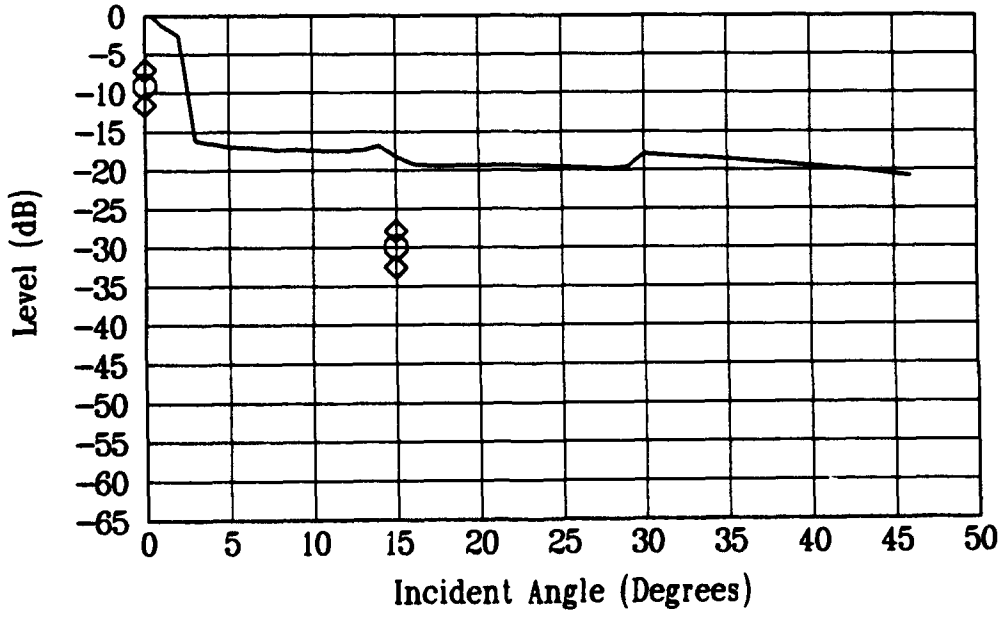

Figure 5.66. $1.53\mu m R_A$ aluminum surface $g \ll 1$ with acoustic λ_o.

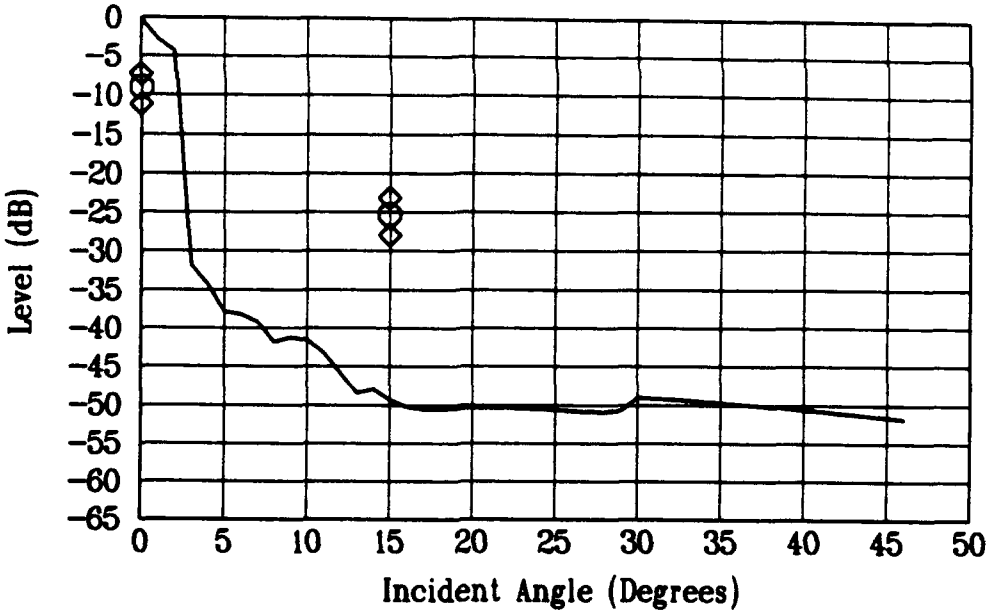

Figure 5.67. $3.05\mu m R_A$ aluminum surface $g \ll 1$ with material λ_o.

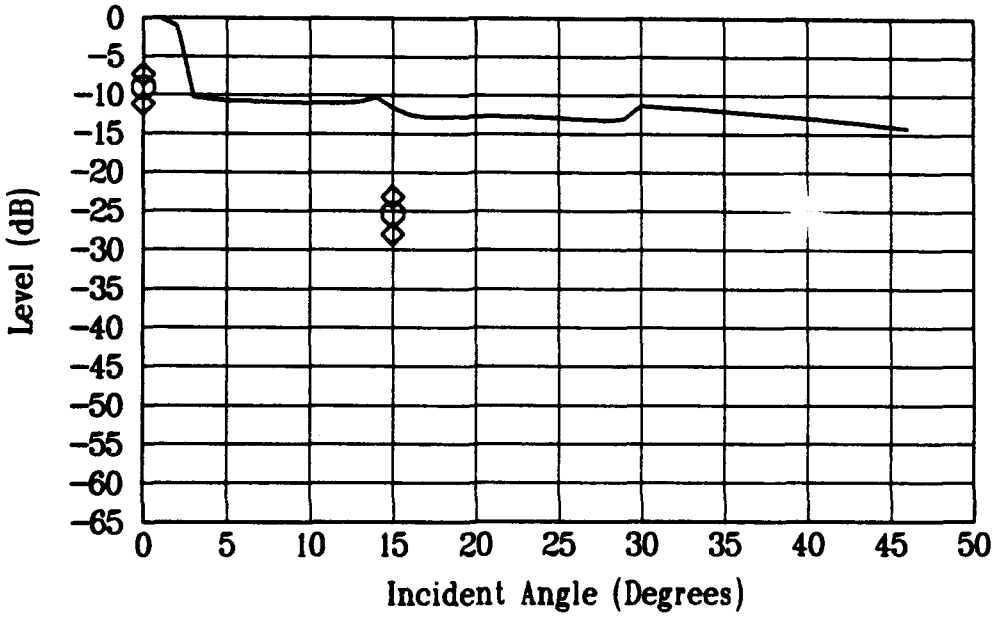

Figure 5.68. $3.05\mu m R_A$ aluminum surface $g \ll 1$ with acoustic λ_o.

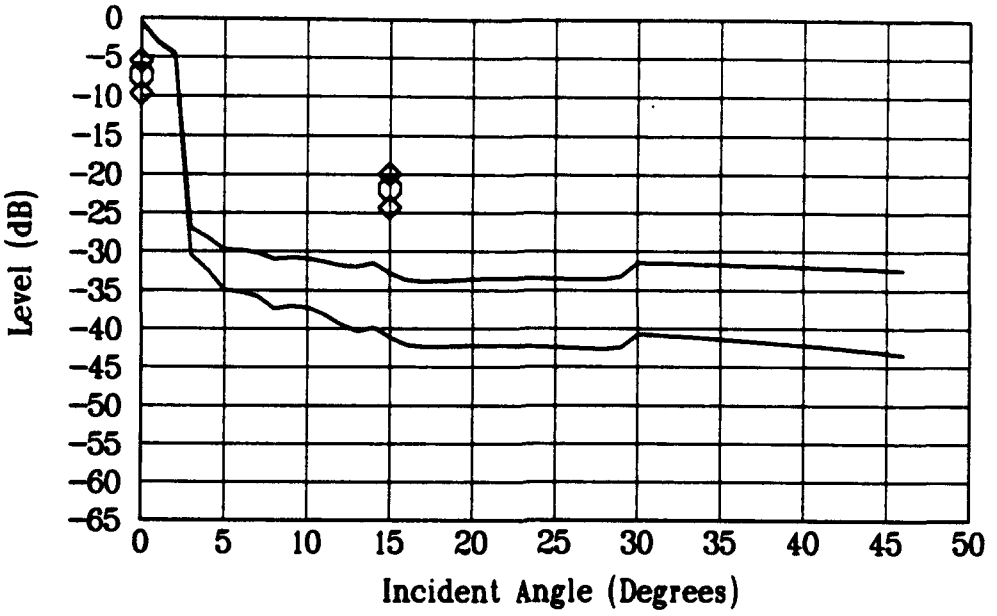

Figure 5.69. 5.08$\mu m R_A$ aluminum surface $g \sim 1$ with material λ_o.

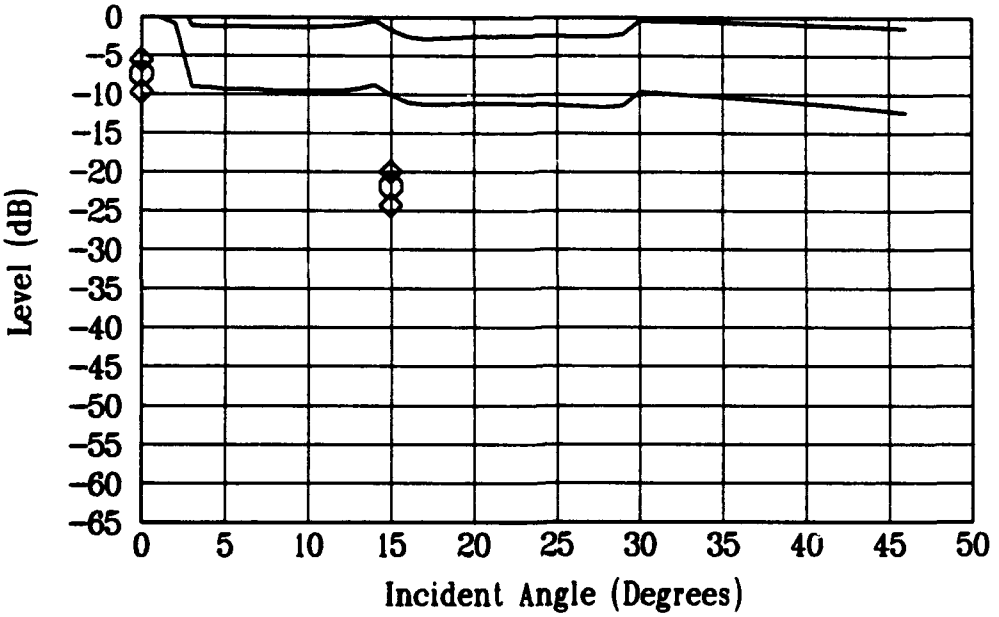

Figure 5.70. 5.08$\mu m R_A$ aluminum surface $g \sim 1$ with acoustic λ_o.

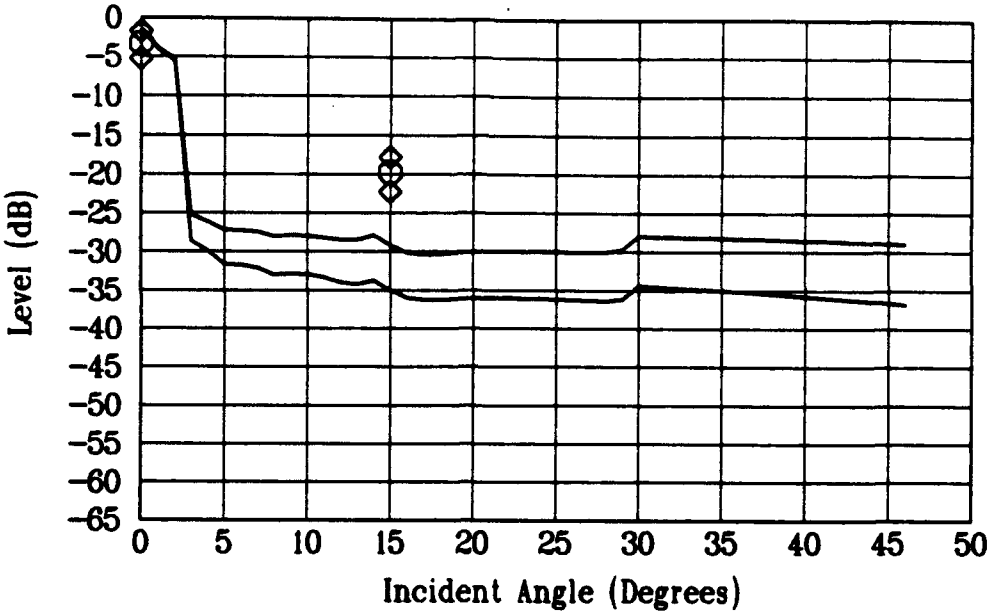

Figure 5.71. $7.62\mu m R_A$ aluminum surface $g \sim 1$ with material λ_o.

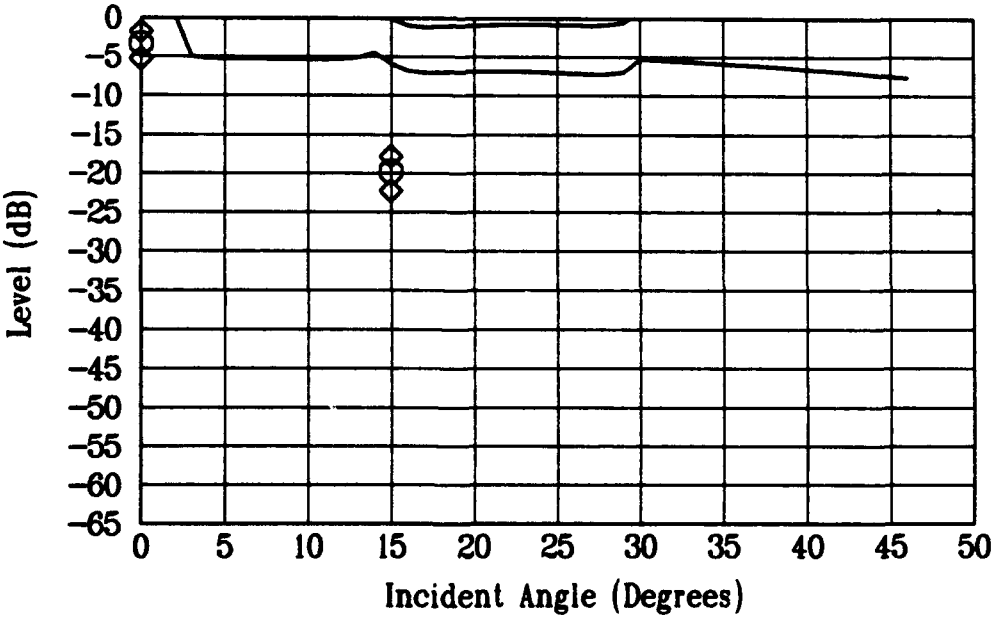

Figure 5.72. $7.62\mu m R_A$ aluminum surface $g \sim 1$ with acoustic λ_o.

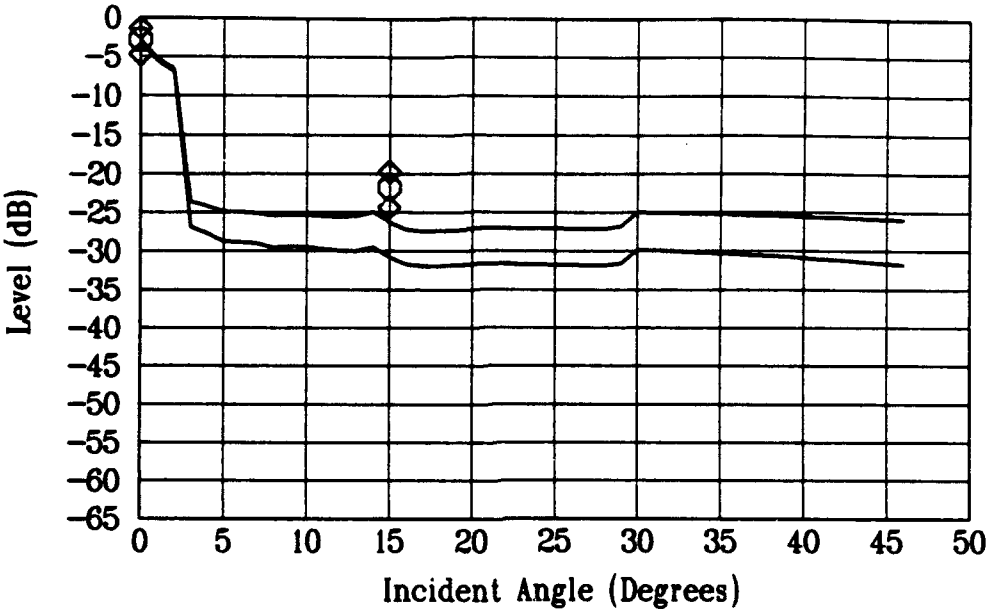

Figure 5.73. $10.67\mu m R_A$ aluminum surface $g \sim 1$ with material λ_o.

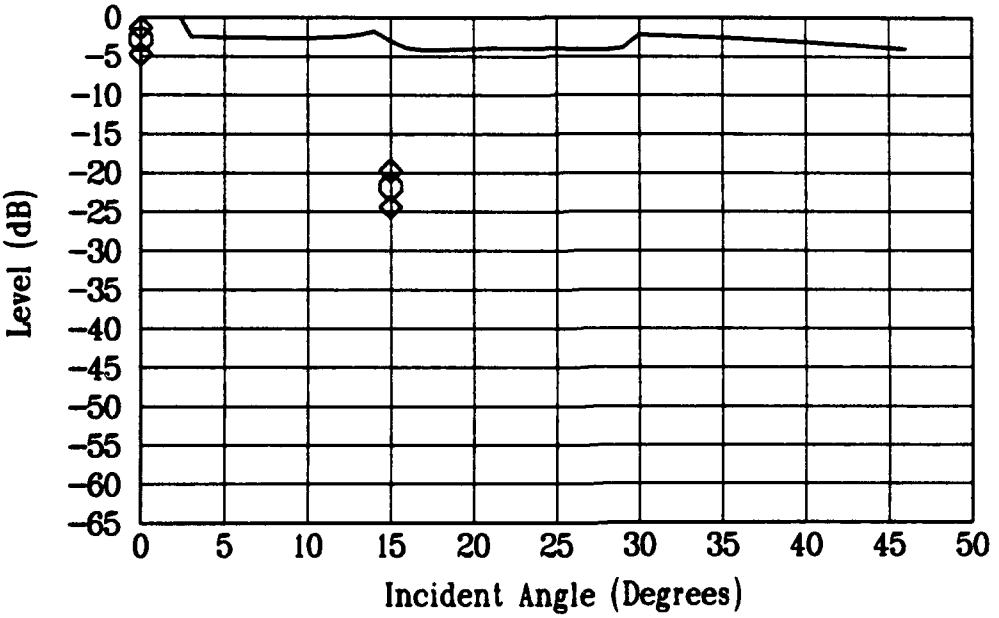

Figure 5.74. $10.67\mu m R_A$ aluminum surface $g \sim 1$ with acoustic λ_o.

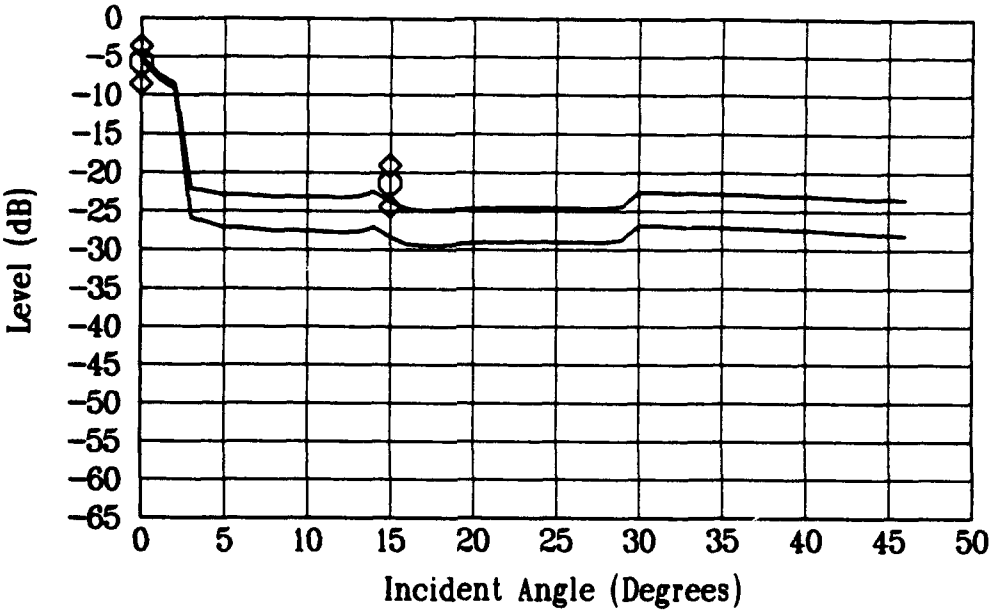

Figure 5.75. $14.22\mu m R_A$ aluminum surface $g \sim 1$ with material λ_o.

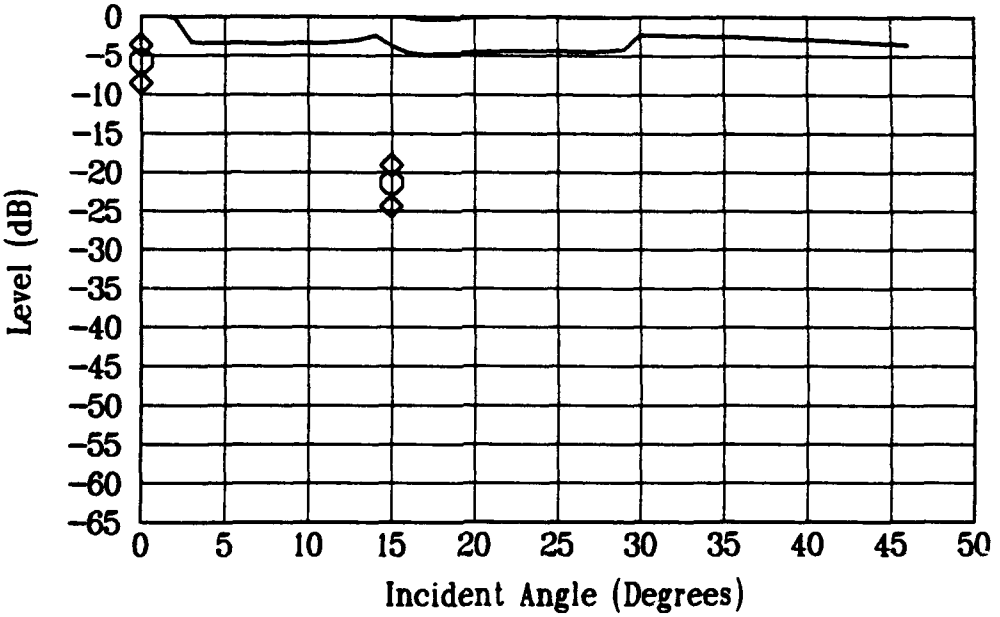

Figure 5.76. $14.22\mu m R_A$ aluminum surface $g \sim 1$ with acoustic λ_o.

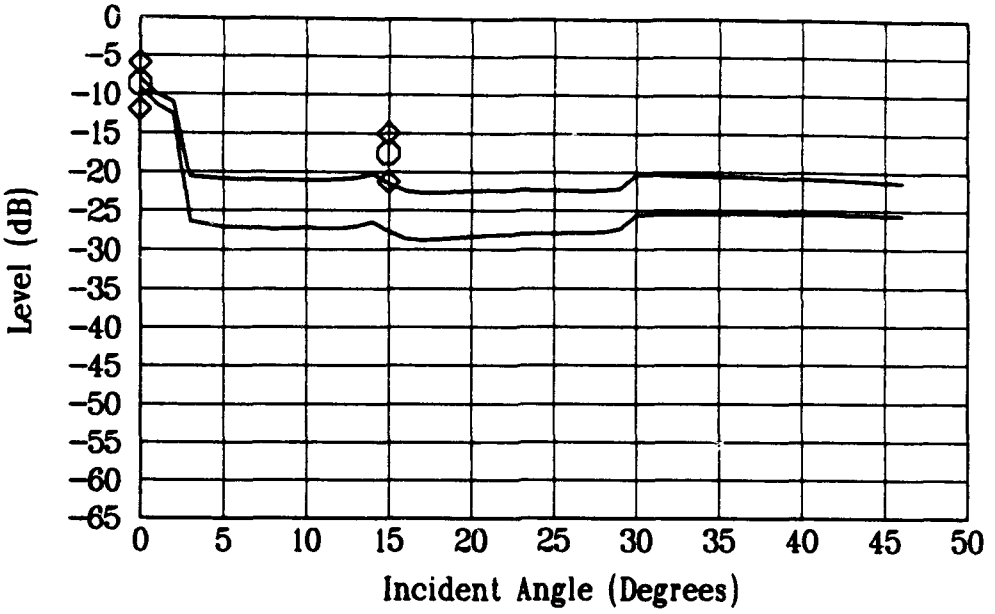

Figure 5.77. $18.29 \mu m R_A$ aluminum surface $g \sim 1$ with material λ_o.

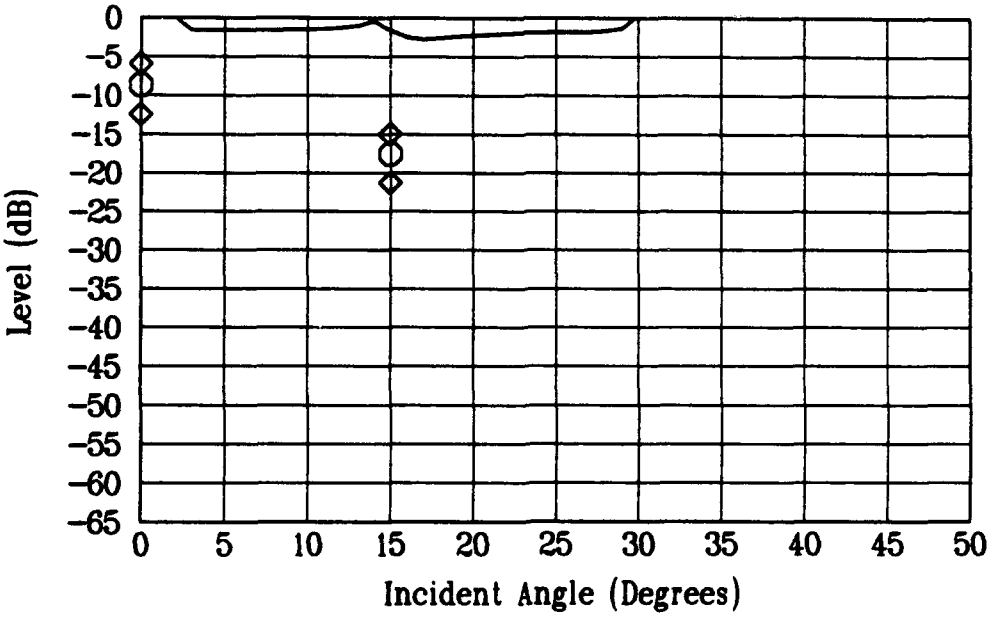

Figure 5.78. $18.29 \mu m R_A$ aluminum surface $g \sim 1$ with acoustic λ_o.

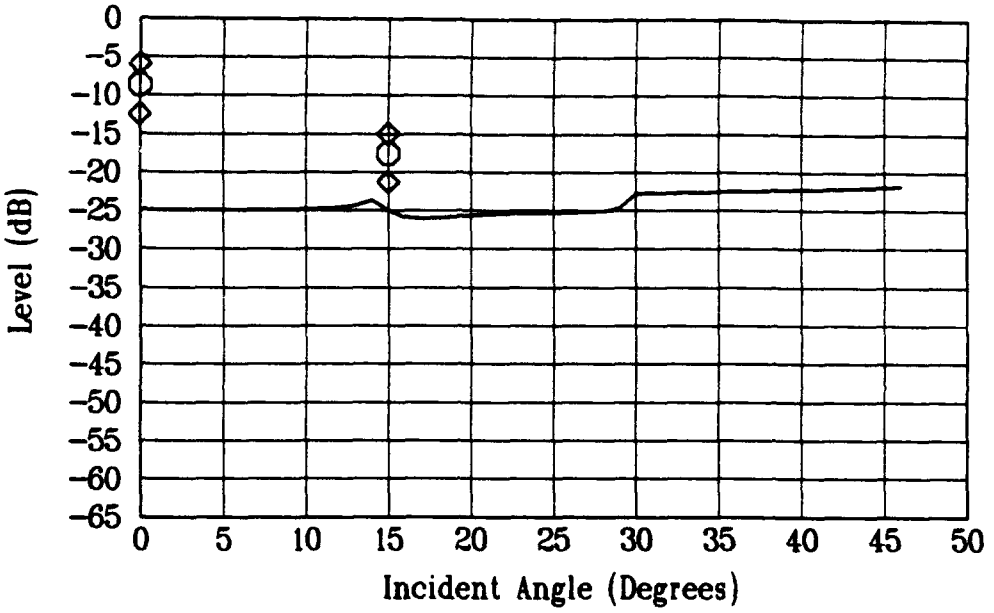

Figure 5.79. $18.29\mu m R_A$ aluminum surface $g \gg 1$ with material λ_o.

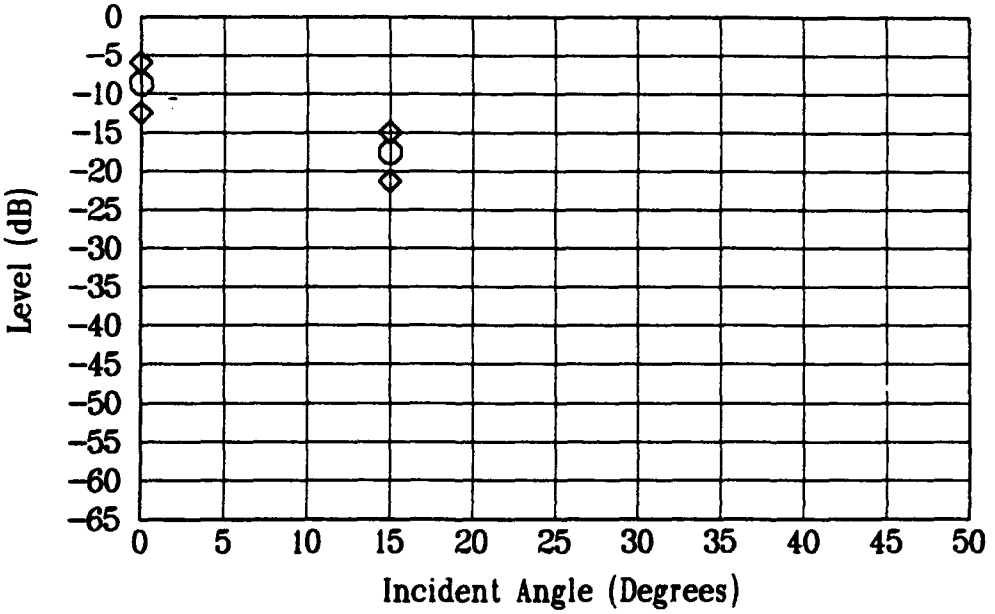

Figure 5.80. $18.29\mu m R_A$ aluminum surface $g \gg 1$ with acoustic λ_o.

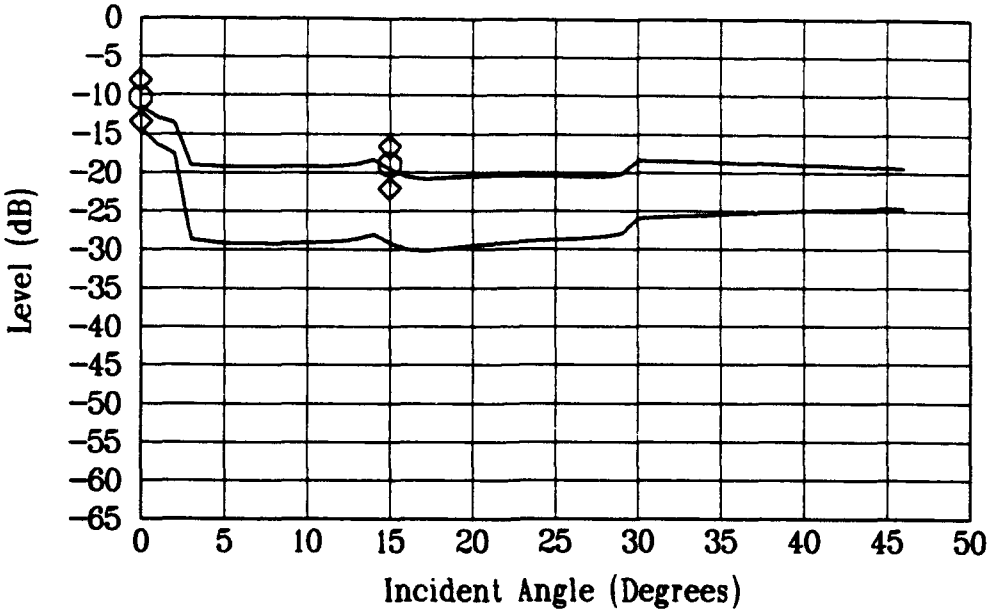

Figure 5.81. $22.86\mu m R_A$ aluminum surface $g \sim 1$ with material λ_o.

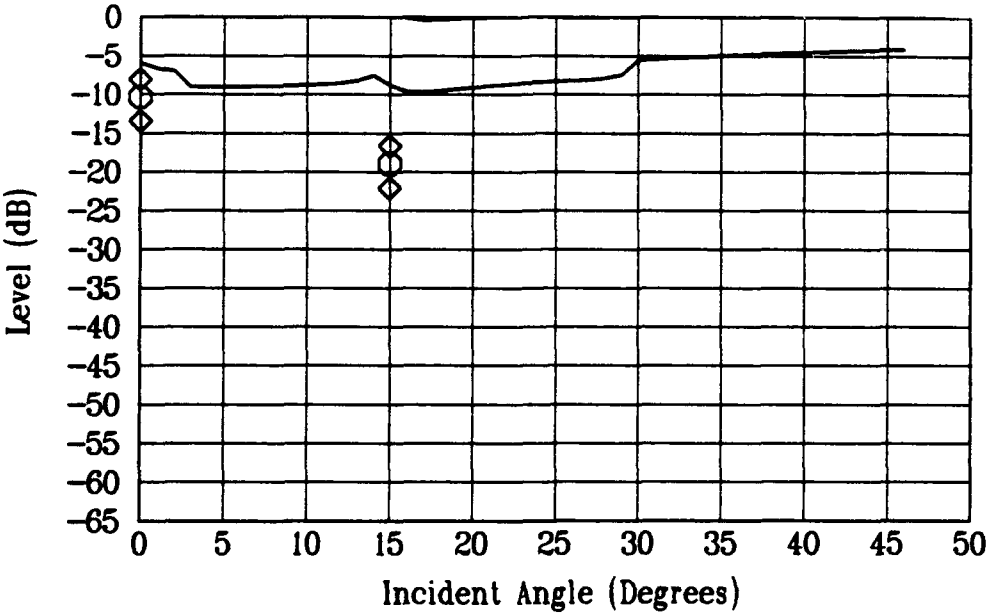

Figure 5.82. $22.86\mu m R_A$ aluminum surface $g \sim 1$ with acoustic λ_o.

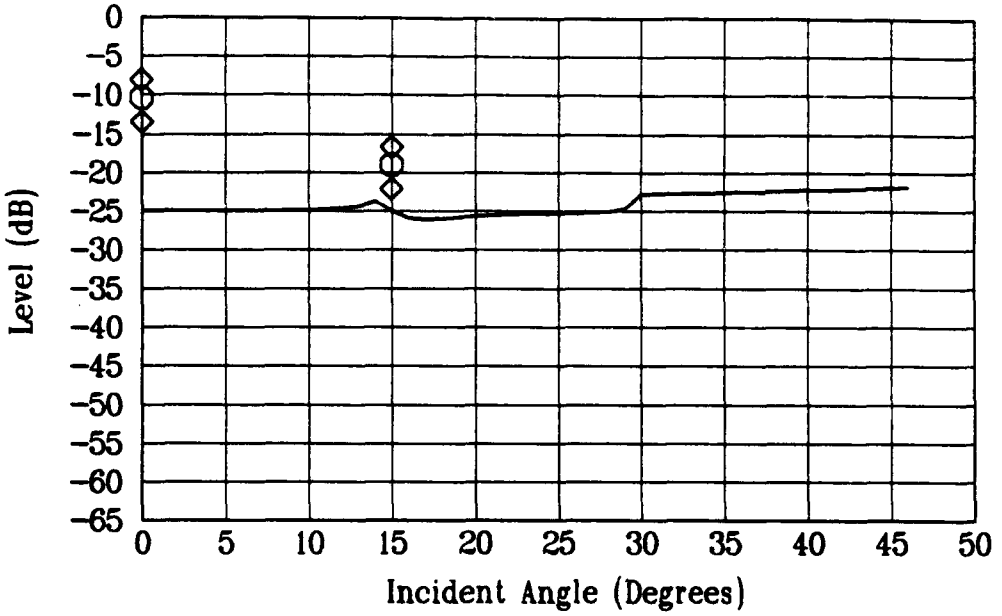

Figure 5.83. 22.86$\mu m R_A$ aluminum surface $g \gg 1$ with material λ_o.

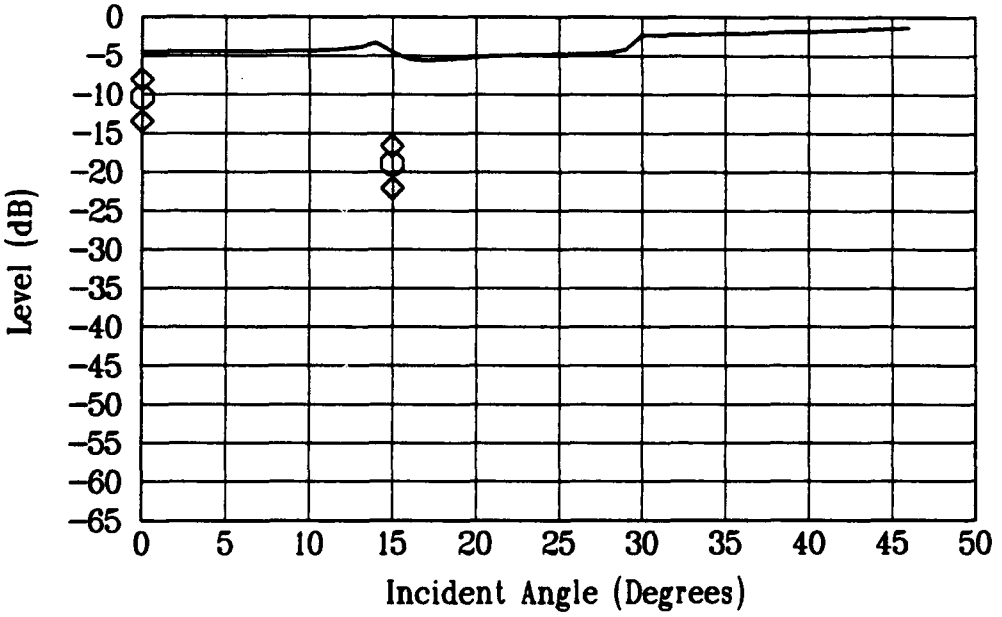

Figure 5.84. 22.86$\mu m R_A$ aluminum surface $g \gg 1$ with acoustic λ_o.

Table 5.8. Material and statistical properties.

R_A μm	g (0°)	λ_o (material) μm	λ_o (acoustic) μm	Material
0.51	$1.81 \cdot 10^{-3}$	2.8	1220	Aluminum
0.58	$2.36 \cdot 10^{-3}$	4.1	1340	Lucite
0.85	$5.07 \cdot 10^{-3}$	5.0	1385	Lucite
1.20	$10.11 \cdot 10^{-3}$	7.5	-	Lucite
1.52	$16.22 \cdot 10^{-3}$	8.3	1180	Aluminum
2.00	$28.07 \cdot 10^{-3}$	9.5	-	Lucite
3.05	$65.29 \cdot 10^{-3}$	16.5	1260	Aluminum
3.50	$85.98 \cdot 10^{-3}$	12.7	-	Lucite
5.08	0.181	27.43	980	Aluminum
7.25	0.369	7.81	1060	Glass
7.62	0.408	41.15	1180	Aluminum
10.67	0.799	57.62	1400	Aluminum
14.22	1.419	76.79	1300	Aluminum
15.23	1.628	16.41	1240	Glass
18.29	2.348	98.77	1980	Aluminum
21.75	3.320	23.44	1270	Glass
22.86	3.668	123.44	1300	Aluminum
31.18	6.823	33.59	1300	Glass
34.37	8.291	37.03	890	Glass

Table 5.9. Scattering level errors for 0°.

R_A μm μm	g	Average Observed Level (dB)	λ_o (material) Error (dB)	λ_o (acoustic) Error (dB)
0.51	($\ll 1$)	-4.3	3.8	4.2
0.58	($\ll 1$)	-3.5	3.0	3.2
0.85	($\ll 1$)	-3.6	2.9	3.2
1.52	($\ll 1$)	-9.1	7.9	9.1
3.05	($\ll 1$)	-9.1	6.8	8.7
5.08	(~ 1)	-7.4	3.7	7.2
7.25	(~ 1)	-9.5	4.2	9.4
7.62	(~ 1)	-3.4	-1.7	3.9
10.67	(~ 1)	-2.9	-3.9	4.1
14.22	(~ 1)	-5.8	-2.9	5.5
15.23	(~ 1)	-7.5	-3.3	6.2
18.29	(~ 1)	-8.7	-1.7	9.0
	($\gg 1$)		-16.2	9.8
21.75	(~ 1)	-9.9	-5.1	4.3
	($\gg 1$)		-29.5	5.2
22.86	(~ 1)	-10.4	-1.5	4.3
	($\gg 1$)		-14.5	5.9
31.18	(~ 1)	-16.4	-3.7	0.8
	($\gg 1$)		-23.0	8.8
34.37	(~ 1)	-19.6	-1.9	0
	($\gg 1$)		-19.8	7.8

Table 5.10. Scattering level errors for 15°.

R_A μm μm	g	Average Observed Level (dB)	λ_o (material) Error (dB)	λ_o (acoustic) Error (dB)
0.51	($\ll 1$)	-26.9	-43.5	-0.2
0.58	($\ll 1$)	-45.3	-26.0	12.8
0.85	($\ll 1$)	-37.5	-32.3	8.6
1.52	($\ll 1$)	-27.9	-31.4	10.0
3.05	($\ll 1$)	-23.2	-25.3	12.9
5.08	(~ 1)	-20.0	-12.4	10.2
7.25	(~ 1)	-27.8	-16.4	20.6
7.62	(~ 1)	-17.9	-11.0	12.3
10.67	(~ 1)	-19.8	-6.2	17.0
14.22	(~ 1)	-19.1	-4.4	15.6
15.23	(~ 1)	-27.4	-10.5	22.9
18.29	(~ 1)	-15.0	-6.3	13.6
	($\gg 1$)		-9.7	15.1
21.75	(~ 1)	-26.5	-8.3	16.4
	($\gg 1$)		-12.6	22.1
22.86	(~ 1)	-16.7	-2.7	8.2
	($\gg 1$)		-8.0	12.4
31.18	(~ 1)	-25.3	-6.4	6.3
	($\gg 1$)		-13.8	17.9
34.37	(~ 1)	-26.7	-4.2	0
	($\gg 1$)		-12.4	15.2

Chapter 6

CONCLUSIONS

The exercise undertaken shows that the technique has the ability to definitely differentiate different levels of roughness. The scan images are the best evidence of the feasibility. The supporting statistical data are not as encouraging. This is not too surprising considering the nature of Kirchhoff theory. The theory is only exact when the surface is a smooth plane (Ogilvy, 1991, p. 100). As the surface becomes more and more rough, errors increase. Still it is the only theory that provides a basis for comparison.

Some, if not all, of the problems seen with the data can be attributed to two problems. The first is the uncertainty of the correct values for correlation length. The various attempts of measuring the value have not been successful. The smallest resolution of the data points using the strip chart yields correlation lengths equal to the sampling interval. The attempt to use the acoustic field has improved the correlation with some of the predicted backscatter values. However, this technique also has some clearly false results with the aluminum cast comparator.

The second problem is the model surfaces. The manually fabricated surfaces using the plexiglas blocks prove to be inconsistent. The primary reason for this is the lack of any standard surfaces in the range of surface roughness of interest. The one manufactured standard, the cast surface comparator, while a standard for roughness, had no documentation for the correlation

length and the surface curvature causes considerable difficulty for measurement. One problem with the bead strips also can not be ignored. The width of the strips was approximately equal to the beamwidth of the transducer. This was not a judicious choice for acquiring a large number of points from a single roughness region.

For continued research in this area, the following suggestions are appropriate. There is a need for reliable standard surfaces that are in the range of surface roughness required. There is also the need for research into the roughness of the lesions themselves. Beyond repeating these same experiments, the next step would be the use of a rotating head transducer similar to the commercial intravascular probe on the interior of cylindrical standard surfaces and in vitro tissue samples.

BIBLIOGRAPHY

Barzilai, B., Saffitz, J. E., Miller, J. G., and Sobel, B. E., "Quantitative Ultrasonic Characterization of the Nature of Atherosclerotic Plaques in Human Aorta," *Circulation Research* **60**, 459-463 (1987).

Bass, F. G., and Fuks, I. M., *Wave Scattering from Statistically Rough Surfaces* (Pergamon Press, Oxford, 1979).

Bendat, J. S., and Piersol, A. G., *Random Data: Analysis and Measurement Procedures* (Wiley-Interscience, New York, 1971).

Bendat, J. S., and Piersol, A. G., *Engineering Applications of Correlation and Spectral Analysis* (Wiley-Interscience, New York, 1980).

Bennett, J. M., and Mattsson, L., *Introduction to Surface Roughness and Scattering* (Optical Society of America, Washington, D. C., 1989).

Bozma, O., and Kuc, R., "Characterizing Pulses Reflected from Rough Surfaces Using Ultrasound," *Journal of the Acoustical Society of America* **89**, 2519-2531 (1991).

Brekhovskikh, L. M., *Waves in Layered Media* (Academic Press, New York, 1960).

Chazov, E. I., and Smirnov, V. N., *Vessel Wall in Athero- and Thrombogenesis*, Ed. Chazov, E. I., and Smirnov, V. N. (Springer-Verlag, Berlin, 1982).

Clay, C. S., and Medwin, H., *Acoustical Oceanography: Principles and Applications* (Wiley-Interscience, New York, 1977).

Coy, K. M., Maurer, G., and Siegel, R. J., "Intravascular Ultrasound Imaging: A Current Perspective.," *Journal of the American College of Cardiology* **18**, 1811-23 (1991).

DeKroon, M. G. M., VanDerWal, L. F., Gussenhoven, W. J., and Bom, N., "Angle-Dependent Backscatter From the Arterial Wall," *Ultrasound in Med. & Biol.* **17**, 121-126 (1991).

DeMaria, A. N., et al., "Intravascular Ultrasound: Where Is It Now and Where Is It Going?," *Heart House Learning Center Highlights*, **Summer**, 17-19 (1990).

Gray, H., *Anatomy, Descriptive and Surgical* (Running Press, Philadelphia, 1974).

IEEE Standard Radar Definitions (Institute of Electrical and Electronic

Engineers, Inc., New York, 1982).

Kinsler, L. E., Frey, A. R., Coppens, A. B., and Sanders, J. V., *Fundamentals of Acoustics* (John Wiley and Sons, New York, 1982).

Likar, I. N., and Robinson, R. W., *Atherosclerosis* (Karger, Basel, 1985).

Lockwood, G. R., Ryan, L. K., Hunt, J. W., and Foster, F. S., "High Frequency Ultrasound Vascular Tissue Characterization," *IEEE 1990 Ultrasound Symposium* , 1409-1412 (1990).

Luke, Y. L., *Integrals of Bessel Functions* (McGraw-Hill, New York, 1962).

McDaniel, S. T., and Gorman, A. D., "Acoustic and Radar Sea Surface Backscatter," *J. of Geophysical Research* 87, 4127-4136 (1982).

National Heart, Lung, and Blood Institute Report of the Task Force on Research in Atherosclerosis (U. S. Department of Health and Human Services, Washington, D. C., 1991).

Norton Specification Manual for Coated Abrasives (Norton Co., Worcester, Ma., 1990).

Officer, C. B., *Introduction to the Theory of Sound Transmission With Application to the Ocean* (McGraw-Hill, New York, 1958).

Ogilvy, J. A., *Theory of Wave Scattering from Random Rough Surfaces* (Adam Hilger, Bristol, 1991).

Papoulis, A., *Probability, Random Variables, and Stochastic Processes* (McGraw-Hill, New York, 1965).

Picano, E., et al., "Angle Dependence of Ultrasonic Backscatter in Arterial Tissues: a Study in Vitro," *Circulation* **72**, 572-576 (1985).

Picano, E., et al., "Different Degrees of Atherosclerosis Detected by Backscattered Ultrasound: An In Vitro Study on Fixed Human Aortic Walls," *J. Clin. Ultrasound* **11**, 375-379 (1983).

Rayleigh, J. W. S., *The Theory of Sound Volume 1* (Dover, New York, 1945).

Rayleigh, J. W. S., *The Theory of Sound Volume 2* (Dover, New York, 1945).

Shung, K. K., "Ultrasonic Characterization of Biological Tissues," *J. Biomechanical Engineering* **107**, 309-314 (1985).

Shung, K. K., Smith, M. B., and Tsui, B. M. W., *Principles of Medical Imaging* (Academic Press, San Diego, 1992).

Shung, K. K., Fei, D. Y., and Bronez, M. A., "Effects of Atherosclerotic Lesions on Ultrasonic Beam and CW Doppler Signals," *J. Clin. Ultrasound* **13**, 11-18 (1985).

Talysurf 10 Operator's Manual (Rank Taylor Hobson, Leicester, England, 1977).

Wells, P. N. T., *Scientific Basis of Medical Imaging* (Churchill Livingstone, Edinburgh, 1982).

Wolverson, M. K., Bashiti, H. M., and Peterson, G. J., "Ultrasonic Tissue Characterization of Atheromatous Plaques Using a High Resolution Real Time Scanner," *Ultrasound in Med. & Biol.* **9**, 599-609 (1983).

Yock, P. G., Fitzgerald, P. J., Linker, D. T., and Angelsen, B. A. J., "Intravascular Ultrasound Guidance for Catheter-Based Coronary Interventions," *Journal of the American College of Cardiology* **17**, 39B-45B (1991).

Appendix

SOFTWARE

This appendix includes a listing of the program REPLOT.CPP. REPLOT is a program that has evolved during the development of this thesis into an all inclusive analysis tool for the scans made with the PDP-11 and the Compaq 386. When the scans are made using the PDP-11 with the Peritek display, the data are stored in a compressed form along with two color maps and a header. To continue using the data it became necessary to develop a program that could redisplay the already collected data on another computer. The data files, on 8" diskettes, were taken to the Electrical Engineering VAX and transferred to $5\frac{1}{4}$" diskettes usable in personal computers. Considerable effort was expended to decode the compression algorithm used by the Peritek board to again display the data. The original data have 256 levels of color or gray scale. The VGA display most commonly used on personal computers has 16 levels of color. The program writes a file of uncompressed data and then shifts the data right by four bits and then displays the data. The original Peritek display is a 512×512 display while a VGA display is 640 wide and 480 high. The data when displayed on the VGA are centered horizontally and the last line of data is aligned with the bottom of the display. This means that 32 lines at the top are omitted. This was chosen because the original scans are at the bottom if they are near an edge. For the scans made later the data have 32 lines of blank data in the beginning in order to align the scan of the aluminum cast surface comparator with the top of the display.

```
// This program takes a file from a Peritek program and          //
// redisplays it If the file already exists in the form required //
// for display on a PC the program will display it and allow     //
// regions of the display to be selected with a mouse and the    //
// mean, standard deviation, and peak value to be found.  It also //
// allows lines, both vertical and horizontal to be selected and //
// the correlation length found along those lines                //
#include <graphics.h>          // For graphics library functions //
#include <stdlib.h>            // For exit()                     //
#include <stdio.h>
#include <conio.h>
#include <alloc.h>
#include <string.h>
#include <math.h>
#include <dos.h>
#include "mouse2.i"
Mresult* Result;
unsigned int plot_peak = 0;
Mstatus PositionR,  PositionL;
float db_ratio, ifloat, average, average_2, st_dev, data_float, n_points,
n_points_off,cor_value, f_data, data_array[1056], corrarray[512];
// Global variables -- set by calc_coords() //
int max_x, max_y, mid_x, mid_y; // Maximum x- and y-coordinates //
unsigned char comp_data, reg_data, map_1, map_2, gflag, data_byte;
unsigned char color_map_upper_word = 0x0F, color_map_lower_word = 0xFF;
unsigned long color_map_1[256], test_value, head4, head5;
unsigned long color_map_2[256];
unsigned char header_words[48];
unsigned char map_full[512],line_scan[512];
char n_boxes[3],x1[4], y1[4], x2[4], y2[4];
unsigned int decode_header(unsigned char w[48], unsigned int word_count);
struct palettetype far *pal_pt, *pal_pt_old=NULL;
struct palettetype far pal, pal_old;
void store_map_1(void);
void store_map_2(void);
void straight_write(void);
void unpack(void);
void statistics(void);
void correl(void);
void linear_gray(struct palettetype pal);
void log_gray(struct palettetype pal);
void old_palette(struct palettetype pal);
void new_palette(struct palettetype pal);
void joe_palette(struct palettetype pal);
int set_graph(void);          // Initialize graphics          //
void calc_coords(void);       // Scale distances onscreen     //
void draw_scan(void);         // Draw scan                    //
void draw_rect(void);         // Draw rectangles from a file  //
void draw_zoom(void);         // Draw a zoom scan             //
void surface_level(void);     // Calculate levels in a region//
void correl_line(void);
void get_key(void);          // Display text on graphics screen//
     // wait for key //
void get_pt1(void);           // Find first point //
void get_pt2(void);           // Find second point //
int peak_detect(unsigned char data_point);  // Find highest level in plot //
unsigned int color_level, head[6];
unsigned int peak_to_now;
int loop_ct, line_ct, array_ind1, scan_x, scan_y,
 EXIT = FALSE, EXIT1 = FALSE;
```

```
int start_x, end_x, start_y, end_y, i_a, j_a, index_cor;
char ans, ans1, b_char = ' ';
char filename[30], filename2[30], outfilename[30], outfilename2[30],
  filename3[30], outfilename3[30];
char inbuf[1];
FILE *file_pt;
FILE *file_pt1;
FILE *file_pt2;
FILE *file_pt3;
FILE *file_pt4;
int upper_left_x, upper_left_y, lower_right_x, lower_right_y, temp_coor;
int del_x,del_y, npoint, boxes, n_box;
int delta_x, delta_y, xmax_limit, ymax_limit;
int TPos( int TP, int Low, int High )
   {
   return( ( TP >= Low ) && ( TP <= High ) );
   }
void BoxItem( int x, int y, int w, int h, char* text )
   {
   settextjustify( CENTER_TEXT, CENTER_TEXT );
   rectangle( x,y,x+w,y+h );
   outtextxy( x+(w/2), y+(h/2),text );
   }
int main (void)
   {
// Find out which *.vch file to be plotted //
   clrscr( );
printf (" Does the data file already exist for plotting on a PC? \n "
" Answer y or Y if it exists. \n");
gets(inbuf);
   sscanf (inbuf,"%1c",&ans);
   if ((ans!='y') && (ans !='Y'))
      {
      printf ("Which VCH data file do you want plotted?   ");
      scanf ("%s",filename);
      if ((file_pt = fopen (filename, "rb")) == NULL)
         {
         fprintf (stderr, "Cannot open input file.");
         return 1;
         }
      printf ("What is the file name for color map output?   ");
      scanf ("%s",outfilename);
      if ((file_pt1 = fopen (outfilename, "wb")) == NULL)
         {
         fprintf (stderr, "Cannot open output file.");
         return 1;
         }
      printf (" What is the file name for data file output?  \n "
              " Use a virtual disk address!!! ");
      scanf ("%s",outfilename2);
      if ((file_pt2 = fopen (outfilename2, "wb")) == NULL)
         {
         fprintf (stderr, "Cannot open output file.");
         return 1;
         }
      if ((file_pt3 = fopen (outfilename3, "wt")) == NULL)
         {
         fprintf (stderr, "Cannot open input file.");
         return 1;
         }
// Start reading the header information in the first three lines  //
// See if first word is '123456'                                  //
```

```
// See if the data is compressed                                    //
// See if the color maps are present                                //
// Store the other parts of the words in the correct parts of       //
// the color map(s) if required //
//                                                                    //
      while (!feof(file_pt))
         {
// Read first three lines //
         fread (header_words,1,48,file_pt);
         clrscr();
         array_ind1=0;
         for (line_ct = 0; line_ct < 3; line_ct++)
            {
            for (loop_ct=0; loop_ct<16; loop_ct++)
               {
               array_ind1 = array_ind1++;
               }
            }
// Function Call //
// Decode first word to be sure it counts 123456          //
// i.e. A72E = 1010|0111|0010|1110 = __1|010|011|100|101|110 //
// = 1,2,3,4,5,6 //
         head[0] = decode_header(header_words,0);
         fprintf (stdout,"head[0] = %x \n ",head[0] );
         /* (head[0] != 0xA72E)
            {
            fprintf (stderr, "File does not have correct header word.");
            return 1;
         }
// Decode second word //
         head[1] = decode_header(header_words,1);
         fprintf (stdout,"head[1] = %x \n ",head[1] );
// Decode third word, beginning x coord //
         head[2] = decode_header(header_words,2);
         fprintf (stdout,"head[2] = %x \n ",head[2] );
// Decode fourth word, beginning y coord //
         head[3] = decode_header(header_words,3);
         fprintf (stdout,"head[3] = %x \n ",head[3] );
// Decode fifth word, ending x coord //
         head[4] = decode_header(header_words,4);
         fprintf (stdout,"head[4] = %x \n ",head[4] );
// Decode sixth word, ending y coord //
         head[5] = decode_header(header_words,5);
         fprintf (stdout,"head[5] = %x \n ",head[5] );
         comp_data = head[1] & 0x80;
         reg_data = head[1] & 0x40;
         map_1 = head[1] & 0x10;
         map_2 = head[1] & 0x20;
         gflag = head[1] & 0x08;
         fprintf (stdout,"%x\t %x\t %x\t %x\t %x \n",gflag,map_1,map_2,
         reg_data, comp_data);
         if (gflag == 0x08) // File contains graphics information //
            {
            if (map_1 == 0x10) // File has top color map //
               {
               store_map_1();
               }
            if (map_2 == 0x20) // File has bottom color map //
               {
               store_map_2();
               }
```

```
            if (reg_data == 0x40) // Data is stored in linear order //
               {
               straight_write();
               }
            if (comp_data == 0x80) // Data is stored in compressed format //
               {
               unpack();
               }
            }
         }
      fclose (file_pt);
      fclose (file_pt1);
      fclose (file_pt2);
      }
   else
      {
      printf (" What is the file name for data file output?  \n "
              " Use a virtual disk address!!! ");
      scanf ("%s",outfilename2);
      }
   if ((file_pt2 = fopen (outfilename2, "rb")) == NULL)
      {
      fprintf (stderr, "Cannot open output file.");
      return 1;
      }
   printf (" What is the file name for correlation data file output?  \n "
           " Use a virtual disk address!!! ");
   scanf ("%s",outfilename3);
   if ((file_pt3 = fopen (outfilename3, "wt")) == NULL)
      {
      fprintf (stderr, "Cannot open output file.");
      return 1;
      }
   printf (" What is the file name for block data rectangles?  \n "
           " Use a virtual disk address!!! ");
   scanf ("%s",filename3);
   if ((file_pt4 = fopen (filename3, "rt")) == NULL)
      {
      fprintf (stderr, "Cannot open input file.");
      return 1;
      }
   fprintf(stdout," Do you wish to see the plot in one of the
                  following formats?\n");
   fprintf(stdout,"       Linear Gray Scale (A)\n");
   fprintf(stdout,"       Log Gray Scale (B)\n");
   fprintf(stdout,"       Color (C)");
   gets(inbuf);
   gets(inbuf);
   sscanf (inbuf,"%1c",&ans);
// Exit if not EGA or VGA //
// Find out if they have what it takes //
   if (set_graph() != 1)
      {
      printf("This program requires EGA or VGA graphics\n");
      exit(0);
      }
   calc_coords();       // Scale to graphics resolution in use //
   draw_scan();      // Draw the scan from the file //
   draw_rect(); // Draw the rectangles around data //
   fprintf(stdout," Find;(a) \n");
   fprintf(stdout," average \n");
```

```
   fprintf(stdout," (b)cor 1 \n");
   gets(inbuf);
   sscanf (inbuf,"%1c",&ans1);
   Result  = gmouse.Mreset();
   if( (ans1 == 'a') || (ans1 == 'A')|| (ans1 == 'b')|| (ans1 == 'B'))
      {
      if( Result->present )
         {
         fprintf (stdprn,"      Data File Name: %s \n ",outfilename2);
         fprintf (stdprn,"      Rectangle File Name: %s \n ",filename3);
         fprintf (stdprn,"      Correlation File Name: %s \n ",outfilename3);
         log_gray(pal);
         setwritemode( COPY_PUT );
         gmouse.Set_Cursor( CIRCLE );
         gmouse.Mshow( FALSE );
         BoxItem( 578, 20, 50, 20, "Pt 1" );
         BoxItem( 578, 50, 50, 20, "Pt 2" );
         BoxItem( 578, 80, 50, 20, "Exit" );
         do
            {
            do   // while ans1 = a or A //
               {
               if ((ans1 == 'a') || (ans1=='A'))
                  {
                  surface_level();   // Pick a region of the display //
                  statistics();
                  fprintf(stdprn,"      AGAIN?  A or N \n");
                  gets( inbuf );
                  sscanf (inbuf,"%1c",&ans1);
                  }
               }while( (ans1 == 'a') || (ans1 == 'A'));
            do   // while ans1 = b or B //
               {
               correl_line();   // Pick a region of the display //
               correl();
               fprintf(stdprn,"      AGAIN?  B or N \n");
               gets( inbuf );
               sscanf (inbuf,"%1c",&ans1);
               }while( (ans1 == 'b') || (ans1 == 'B'));
            }while( (ans1 == 'a') || (ans1 == 'A') ||
(ans1 == 'b') || (ans1 == 'B'));
         }
      }
   get_key();           // Display message and wait for key press //
   closegraph();        // Close graphics system //
   fclose (file_pt2);
   fclose (file_pt3);
   fclose (file_pt4);
   return 0;
   }  // End of main //
//*************************************************************************//
unsigned int decode_header(unsigned char w[48], unsigned int word_count)
   {
   unsigned char header_word = 0xF0;
   unsigned int part1, part2, part3, part4, i1, i2, i3, i4;

// The indexes are reversed because the segments of words are placed in the  //
// header in reversed order //
   i1 = word_count * 8 + 7;
   i2 = word_count * 8 + 5;
   i3 = word_count * 8 + 3;
   i4 = word_count * 8 + 1;
```

```
   part1 = (w[i1] & header_word)<<8;
   part2 = (w[i2] & header_word)<<4;
   part3 = (w[i3] & header_word);
   part4 = (w[i4] & header_word)>>4;
   return (part1 | part2 | part3 | part4);
   }
//*****************************************************************//
//*****************************************************************//
void store_map_1(void)
   {
   unsigned char header_mask1 = 0x0F;
   unsigned char header_mask2 = 0xF0;
   unsigned char part1, part2, part3;
   unsigned long partA, partB, partC;
   unsigned int i1;
// Enter the data from the fir t 48 lower words into the   //
// first 24 color map words /
   for (i1 = 0; i1 < 24; i1++)
      {
      partB = (header_mask2 & header_words[i1*2]);  // Lower Green //
      partC = (header_mask1 & header_words[i1*2]);  // Lower Blue //
      partA = (header_mask1 & header_words[i1*2+1]);  // Lower Red //
      color_map_1[i1] =((partA<<16) | (partB<<4) | partC);
      }
// Get the rest of the 512 lower words and convert them to the     //
// lower 256 color map 1 values //
   rewind(file_pt);
   fread (map_full,1,512,file_pt);
   array_ind1=0;
   for (line_ct = 0; line_ct < 32; line_ct++)
      {
      for (loop_ct=0; loop_ct<16; loop_ct++)
         {
         array_ind1 = array_ind1++;
         }
      }
   for (i1 = 24; i1 < 256; i1++)
      {
      partB = (header_mask2 & map_full[i1*2]);  // Lower Green //
      partC = (header_mask1 & map_full[i1*2]);  // Lower Blue //
      partA = (header_mask1 & map_full[i1*2+1]);  // Lower Red //
      color_map_1[i1] =((partA<<16) | (partB<<4) | partC);
      }
// Get the 512 upper words and convert them to the  //
// upper 256 color map 1 values //
   fread (map_full,1,512,file_pt);
   array_ind1=0;
   for (line_ct = 0; line_ct < 32; line_ct++)
      {
      for (loop_ct=0; loop_ct<16; loop_ct++)
         {
         array_ind1 = array_ind1++;
         }
      }
   for (i1 = 0; i1 < 256; i1++)
      {
      partB = (header_mask2 & map_full[i1*2]);  // Upper Green //
      partC = (header_mask1 & map_full[i1*2]);  // Upper Blue //
      partA = (header_mask1 & map_full[i1*2+1]);  // Upper Red //
      color_map_1[i1] =((partA<<20) | (partB<<8) | (partC<<4) | color_map_1[i1]);
      fprintf (file_pt1,"%c",(0xff000000 & color_map_1[i1])>>24);
```

```
      fprintf (file_pt1,"%c",(0x00ff0000 & color_map_1[i1])>>16);
      fprintf (file_pt1,"%c",(0x0000ff00 & color_map_1[i1])>>8);
      fprintf (file_pt1,"%c",(0x000000ff & color_map_1[i1]));
      }
   }
//*********************************************************************//
void store_map_2(void)
   {
   unsigned char header_mask1 = 0x0F;
   unsigned char header_mask2 = 0xF0;
   unsigned char part1, part2, part3;
   unsigned long partA, partB, partC;
   unsigned int i1;

// Get the 512 lower words and convert them to the    //
// lower 256 color map 2 values //
   fread (map_full,1,512,file_pt);
   array_ind1=0;
   for (line_ct = 0; line_ct < 32; line_ct++)
      {
      for (loop_ct=0; loop_ct<16; loop_ct++)
         {
         array_ind1 = array_ind1++;
         }
      }
   for (i1 = 0; i1 < 256; i1++)
      {
      partB = (header_mask2 & map_full[i1*2]);  // Lower Green //
      partC = (header_mask1 & map_full[i1*2]);  // Lower Blue //
      partA = (header_mask1 & map_full[i1*2+1]);  // Lower Red //
      color_map_2[i1] =((partA<<16) | (partB<<4) | partC);
      }
// Get the 512 upper words and convert them to the  //
// upper 256 color map 2 values //
   fread (map_full,1,512,file_pt);
   array_ind1=0;
   for (line_ct = 0; line_ct < 32; line_ct++)
      {
      for (loop_ct=0; loop_ct<16; loop_ct++)
         {
         array_ind1 = array_ind1++;
         }
      }
   for (i1 = 0; i1 < 256; i1++)
      {
      partB = (header_mask2 & map_full[i1*2]);  // Upper Green //
      partC = (header_mask1 & map_full[i1*2]);  // Upper Blue //
      partA = (header_mask1 & map_full[i1*2+1]);  // Upper Red //
      color_map_2[i1] =((partA<<20) | (partB<<8) | (partC<<4) | color_map_2[i1]);
      fprintf (file_pt1,"%c",(0xff000000 & color_map_2[i1])>>24);
      fprintf (file_pt1,"%c",(0x00ff0000 & color_map_2[i1])>>16);
      fprintf (file_pt1,"%c",(0x0000ff00 & color_map_2[i1])>>8);
      fprintf (file_pt1,"%c",(0x000000ff & color_map_2[i1]));
      }
   }
// *********************************************************************//
void straight_write(void)
   {
// Data is recorded linearly in the *.vch file.                    //
// Read each byte and write the byte to the file_pt2 file.         //
// The data will be organized in 512 lines with each line having   //
```

```
// 512 data bytes. //
   array_ind1=0;
   for (line_ct = 0; line_ct < 512; line_ct++)
      {
      for (loop_ct=0; loop_ct<512; loop_ct++)
         {
         fread (&data_byte,sizeof(unsigned char),1,file_pt);
         fprintf (file_pt2,"%c",data_byte);
         }
      }
   }
//*********************************************************************//
void unpack(void)
// Data is compressed in the *.vch file.      //
// Read a byte and decode the instruction and the count.                 //
// If the word says number continued read rest of number in the next byte.//
// If the next data point represents n points write n points.            //
// If the next data point represents the first of n points,              //
// read and write n points.                                              //
// The data will be organized in a continuous line. //
   {
   unsigned long total_count = 0, total_points;
   unsigned char seq_test = 0x80, count_test = 0x40, upper_mask = 0x1F,
   lower_mask = 0xFF, seq_result;
   unsigned int x_count = 0, y_count = 0, seq_ct, stream_ct;
   int index;
   head4 = head[4]+1-head[2];
   head5 = head[5]+1-head[3];
   total_points = head4 * head5;

   do
      {
      fread (&data_byte,sizeof(unsigned char),1,file_pt);
      seq_result = data_byte & seq_test;
      if (seq_result == 0x80)  // A lot of points are stored in one # //
         {
         seq_ct = upper_mask & data_byte; // # of points //
         if ((count_test & data_byte) == count_test) // Count cont'd? //
            {
//  More count in next word; Read it and shift the   //
// higher bits to the left and add in the second    //
// word //
            fread (&data_byte,sizeof(unsigned char),1,file_pt);
            seq_ct = (seq_ct << 8) | (lower_mask & data_byte);
            }
         fread (&data_byte,sizeof(unsigned char),1,file_pt);
// Store that data word in seq_ct points //
         for (index=0; index<seq_ct; index++)
            {
            fprintf (file_pt2,"%c",data_byte);
            }
         total_count += seq_ct;
         }
      else  // A number of different values are next //
         {
         stream_ct = upper_mask & data_byte; // # of points //
         if ((count_test & data_byte) == count_test) // Count cont'd? //
            {
//  More count in next word; Read it and shift the  //
// higher bits to the left and add in the second   //
// word //
```

```
            fread (&data_byte,sizeof(unsigned char),1,file_pt);
            stream_ct = (stream_ct << 8) | (lower_mask & data_byte);
            }
//  Store that data word in stream_ct points //
         for (index=0; index<stream_ct; index++)
            {
            fread (&data_byte,sizeof(unsigned char),1,file_pt);
            fprintf (file_pt2,"%c",data_byte);
            }
         total_count += stream_ct;
         }
      } while (total_count < total_points);
   y_count = total_count / 512;
   x_count = total_count % 512;
   fprintf (stdout,"x_count = %3d \t y_count =  %3d",x_count,y_count);
   }
//*********************************************************************//

//*********************************************************************//

int set_graph(void)
   {
   int graphdriver = DETECT, graphmode, error_code;

// Initialize graphics system; must be EGA or VGA //
   initgraph(&graphdriver, &graphmode, "");
   error_code = graphresult();
   if (error_code != grOk)
      return(-1);               // No graphics hardware found //
   if ((graphdriver != EGA) && (graphdriver != VGA))
      {
      closegraph();
      return 0;
      }
   return(1);                   // Graphics OK, so return "true" //
   }
//*********************************************************************//

void calc_coords(void)
   {
// Set global variables for drawing //
   max_x = getmaxx();           // Returns maximum x-coordinate //
   max_y = getmaxy();           // Returns maximum y-coordinate //
   }
//*********************************************************************//

void draw_scan(void)
   {
// initialize graphics and local variables //
// of the setrgbpalette function //

   int gdriver = DETECT , gmode, errorcode, errorcode1;
   initgraph(&gdriver, &gmode,"");
   errorcode = graphresult();
   if (errorcode != grOk)
      {
      fprintf(stdout,"Graphics error: %s\n",grapherrormsg(errorcode));
      fprintf(stdout,"Press any key to halt:");
      getch();
      exit(1);
      }
```

```
    getpalette(&pal);
    getpalette(&pal_old);
    pal_pt_old = getdefaultpalette();
 // create gray scale //
    if ((ans == 'a') || (ans == 'A'))
       linear_gray(pal);
    if ((ans == 'b') || (ans == 'B'))
       log_gray(pal);
    max_x = getmaxcolor();
    for (scan_y = 0; scan_y < 512; scan_y++)
       {
       fread (line_scan,sizeof(unsigned char),512,file_pt2);
       for (scan_x=0; scan_x<512; scan_x++)
          {
          data_byte = line_scan[scan_x];
          plot_peak = peak_detect(data_byte);
          if(scan_y>31)
             {
             putpixel(scan_x + 64, scan_y - 32, data_byte>>4);
             }
          }
       }
 // Redisplay ??????? //
    fprintf(stdout,"Peak=\n%3d\n",plot_peak);

    while((ans=='a')||(ans=='b')||(ans=='c')||(ans=='A')||(ans=='B')||
          (ans=='C')||(ans=='d')||(ans=='D')||(ans=='j')||(ans=='J')||
          (ans=='r')||(ans=='R')||(ans=='z')||(ans=='Z'))
       {
       outtextxy(0,110, "Redo?");
       gets(inbuf);
       sscanf (inbuf,"%1c",&ans);
       if ((ans == 'a') || (ans == 'A'))
          linear_gray(pal);
       else if ((ans == 'b') || (ans == 'B'))
          log_gray(pal);
       else if ((ans == 'c') || (ans == 'C'))
          {
          old_palette(pal_old);
          }
       else if ((ans == 'd') || (ans == 'D'))
          {
          new_palette(pal_old);
          }
       else if ((ans == 'r') || (ans == 'R'))
          {
          draw_rect();
          }
       else if ((ans == 'j') || (ans == 'J'))
          {
          joe_palette(pal_old);
          }
       else if ((ans == 'z') || (ans == 'Z'))
          {
          draw_zoom();
          }
       }
    }

//*************************************************************************//
void get_key(void)
   {
```

```
   printf (" Press \n "
           " any key \n"
           " to exit \n");
   getch();
   }
//*************************************************************************//
void linear_gray(struct palettetype pal)
   {
   int i, ired, igreen, iblue;
   setactivepage(0);
   for (i=0; i<pal.size; i++)
      {
      ired = i*4;
      igreen = i*4;
      iblue = i*4;
      setrgbpalette(pal.colors[i], ired, igreen, iblue);
      }
   }
//*************************************************************************//
   void log_gray(struct palettetype pal)
   {
   int i, ired, igreen, iblue;
   float ifloat, db_ratio;
// Calculate a log scale for the gray scale //
   setactivepage(0);
   db_ratio = 63.0 / log10(pal.size+1);
   for (i=0; i<pal.size; i++)
      {
      ifloat = i+1;
      ired = (db_ratio * log10(ifloat) + 0.5);
      igreen = ired;
      iblue = ired;
      setrgbpalette(pal.colors[i], ired, igreen, iblue);
      }
   }
//*************************************************************************//
int peak_detect(unsigned char data_point)   // Find highest level in plot //
   {
   if (data_point>peak_to_now)
peak_to_now = data_point;
   return peak_to_now;
   }
//*************************************************************************//
void old_palette(struct palettetype pal)
   {
   setactivepage(1);
   setallpalette(&pal);
   setallpalette(&pal);
   // color palette         red green blue  //
   setrgbpalette(pal.colors[0], 0 , 0 , 0);
   setrgbpalette(pal.colors[1], 0 , 0 , 63);
   setrgbpalette(pal.colors[2], 0 , 63 , 0);
   setrgbpalette(pal.colors[3], 0 , 63 , 63);
   setrgbpalette(pal.colors[4], 63 , 0 , 0);
   setrgbpalette(pal.colors[5], 63 , 0 , 63);
   setrgbpalette(pal.colors[6], 44 , 44 , 44);
   setrgbpalette(pal.colors[7], 44 , 21 , 0);
   setrgbpalette(pal.colors[8], 21 , 21 , 21);
   setrgbpalette(pal.colors[9], 0 , 0 , 32);
   setrgbpalette(pal.colors[10], 0 , 32 , 0);
   setrgbpalette(pal.colors[11], 0 , 32 , 32);
```

```
   setrgbpalette(pal.colors[12], 32 , 0 , 0);
   setrgbpalette(pal.colors[13], 32 , 0 , 32);
   setrgbpalette(pal.colors[14], 32 , 32 , 0);
   setrgbpalette(pal.colors[15], 63 , 63 , 63);
   }
//*************************************************************************//
void new_palette(struct palettetype pal)
   {
   setactivepage(1);
   setallpalette(&pal);
// color palette         red green blue  //
   setrgbpalette(pal.colors[0], 0 , 0 , 0);
   setrgbpalette(pal.colors[1], 32 , 32 , 32);
   setrgbpalette(pal.colors[2], 63 , 52 , 63);
   setrgbpalette(pal.colors[3], 63 , 0 , 52);
   setrgbpalette(pal.colors[4], 63 , 0 , 20);
   setrgbpalette(pal.colors[5], 63 , 12 , 0);
   setrgbpalette(pal.colors[6], 63 , 44 , 0);
   setrgbpalette(pal.colors[7], 44 , 63 , 0);
   setrgbpalette(pal.colors[8], 12 , 63 , 0);
   setrgbpalette(pal.colors[9], 0 , 63 , 24);
   setrgbpalette(pal.colors[10], 0 , 63 , 56);
   setrgbpalette(pal.colors[11], 0 , 32 , 63);
   setrgbpalette(pal.colors[12], 0 , 0 , 63);
   setrgbpalette(pal.colors[13], 32 , 0 , 63);
   setrgbpalette(pal.colors[14], 52 , 0 , 52);
   setrgbpalette(pal.colors[15], 20 , 0 , 20);
   }
//*************************************************************************//
void joe_palette(struct palettetype pal)
   {
   setactivepage(1);
   setallpalette(&pal);
   setpalette(0,21);
   setpalette(1,22);
   setpalette(2,23);
   setpalette(3,24);
   setpalette(4,25);
   setpalette(5,26);
   setpalette(6,27);
   setpalette(7,28);
   setpalette(8,29);
   setpalette(9,30);
   setpalette(10,31);
   setpalette(11,32);
   setpalette(12,33);
   setpalette(13,34);
   setpalette(14,35);
   setpalette(15,36);
   }
//*************************************************************************//
void draw_zoom(void)
   {
   int dummy_x = 0, dummy_y = 0;
   outtextxy(0,140, "X START?");
   gets(inbuf);
   sscanf (inbuf,"%d",&start_x);
   outtextxy(0,170, "Y START?");
   gets(inbuf);
   sscanf (inbuf,"%d",&start_y);
```

```
   if (start_x < 0)
      start_x = 0;
   if (start_y < 0)
      start_y = 0;
   if (start_x >= 256)
      start_x = 255;
   end_x = start_x + 256;
   if (start_y >= 256)
      start_y = 255;
   end_y = start_y + 256;
   rewind(file_pt2);
   for (scan_y = 0; scan_y < 512; scan_y++)
      {
      fread (line_scan,sizeof(unsigned char),512,file_pt2);
      for (scan_x=0; scan_x<512; scan_x++)
         {
         data_byte = line_scan[scan_x];
         plot_peak = peak_detect(data_byte);
         if((scan_y>=start_y)&&(scan_y<=end_y))
            {
            if((scan_x>=start_x)&&(scan_x<=end_x))
               {
               dummy_x = 2 * (scan_x - start_x);
               dummy_y = 2 * (scan_y - start_y);
               putpixel((dummy_x + 64),(dummy_y-32),data_byte>>4);
               putpixel((dummy_x + 65),(dummy_y-32),data_byte>>4);
               putpixel((dummy_x + 64),(dummy_y-31),data_byte>>4);
               putpixel((dummy_x + 65),(dummy_y-31),data_byte>>4);
               }
            }
         }
      }
// ************************************************************************ //

// ************************************************************************ //
void surface_level(void)
   {
// Establish the cursor //
   Result  = gmouse.Mreset();
   setwritemode( COPY_PUT );
   gmouse.Set_Cursor( CIRCLE );
   gmouse.Mshow( FALSE );
   gmouse.Mshow( TRUE );
   gmouse.Mxlimit( 64, 628 );
   EXIT = FALSE;
   do                // while( EXIT != TRUE )
      {
      PositionR = gmouse.Mpressed( ButtonR );
// Cycle through selection until satisfied //
      if( TPos( PositionR.xaxis, 578, 628 ) )
         {
         if( TPos( PositionR.yaxis, 20, 40 ) )
            get_pt1();
         if( TPos( PositionR.yaxis, 50, 70 ) )
            get_pt2();
         if( TPos( PositionR.yaxis, 80, 100 ) )
            {
            EXIT = TRUE;
//                Draw a box around the area //
            gmouse.Mshow( FALSE );
            if(upper_left_x > lower_right_x )
```

```
                {
                temp_coor = upper_left_x;
                upper_left_x = lower_right_x;
                lower_right_x = temp_coor;
                }
             if(upper_left_y > lower_right_y )
                {
                temp_coor = upper_left_y;
                upper_left_y = lower_right_y;
                lower_right_y = temp_coor;
                }
                rectangle( upper_left_x, upper_left_y,
                   lower_right_x, lower_right_y );
             }
          }
       } while( EXIT != TRUE );
   }
// ***************************************************************
void get_pt1( void)
   {
// Select the first point //
   gmouse.Mxlimit( 64, 575 );
   do                //  while(PositionL.button_status==0 )
      {
      PositionL = gmouse.Mpressed( ButtonL );
      if( PositionL.button_status==1 )
         {
         upper_left_x = PositionL.xaxis;
         upper_left_y = PositionL.yaxis;
         gmouse.Mshow( FALSE );
         putpixel(upper_left_x, upper_left_y, 0);
         gmouse.Mshow( TRUE );
         }
      }while(PositionL.button_status==0 );
   gmouse.Mxlimit( 64, 628 );
   }
// ***************************************************************
void get_pt2( void)
   {
// Select the second point //
   gmouse.Mxlimit( 64, 575 );
   do               // while(PositionL.button_status== 0)
      {
      PositionL = gmouse.Mpressed( ButtonL );
      if( PositionL.button_status==1 )
         {
         lower_right_x = PositionL.xaxis;
         lower_right_y = PositionL.yaxis;
         gmouse.Mshow( FALSE );
         putpixel(lower_right_x, lower_right_y, 0);
         gmouse.Mshow( TRUE );
         }
      }while(PositionL.button_status== 0);
      gmouse.Mxlimit( 64, 628 );
   }
//*************************************************************************//
void draw_rect(void)
   {
// initialize graphics and local variables //
// of the setrgbpalette function //
   fread (&n_boxes,sizeof(unsigned char),3,file_pt4);
   boxes= atol(n_boxes);
```

```
   for (n_box = 0; n_box < boxes; n_box++)
      {
      fread (&x1,sizeof(unsigned char),4,file_pt4);
      fread (&y1,sizeof(unsigned char),4,file_pt4);
      fread (&x2,sizeof(unsigned char),4,file_pt4);
      fread (&y2,sizeof(unsigned char),4,file_pt4);
      upper_left_x = atoi(x1);
      upper_left_y = atoi(y1);
      lower_right_x = atoi(x2);
      lower_right_y = atoi(y2);
      rectangle( upper_left_x, upper_left_y,
         lower_right_x, lower_right_y );
      }
   }
// ************************************************************
void statistics(void)
// Rewind the data file and find the average value of the level //
   {
   fprintf (stdprn,"      X1 = %d Y1 = %d X2 = %d Y2 = %d \n ",upper_left_x,
upper_left_y, lower_right_x, lower_right_y );
   xmax_limit = lower_right_x - 64;
   ymax_limit = lower_right_y + 32;
   average = 0;
   average_2 = 0;
   st_dev = 0;
   n_points = 0;
   n_points_off = 0;
   plot_peak = 0;
   peak_to_now = 0;
   rewind(file_pt2);
   for (scan_y = 0; scan_y <= ymax_limit; scan_y++)
      {
      fread (line_scan,sizeof(unsigned char),512,file_pt2);
      if( (scan_y>=(upper_left_y+32)) && (scan_y<=(lower_right_y+32)) )
         {
         for (scan_x=0; scan_x<=xmax_limit; scan_x++)
            {
            if( (scan_x>=(upper_left_x - 64)) &&
               (scan_x<=(lower_right_x - 64)) )
               {
               data_byte = line_scan[scan_x];
               data_float = data_byte;
               plot_peak = peak_detect(data_byte);
               average = average + data_float;
               average_2 = average_2 + (data_float * data_float);
               n_points = n_points + 1;
               }
            }
         }
      }
   average = average / n_points;
   average_2 = ( average_2 - n_points * average * average );
// Find the standard deviation //

   if( average_2 <= 0 )
      {
      st_dev = 0.0;
      }
   else
      {
      st_dev = sqrt( ( average_2 / n_points ) );
      }
```

```
// Print this information out //
   fprintf (stdprn,"      Number of Points = %6.0f \n ",n_points );
   fprintf (stdprn,"      Peak = %d \n ",plot_peak );
   fprintf (stdprn,"      Mean = %6.2f \n ",average );
   fprintf (stdprn,"      Standard Deviation = %6.2f \n ",st_dev );
   }
// ************************************************************
void correl(void)
   {
// Rewind the data file and find the correlation length of the data //
   fprintf (stdprn,"       X1 = %d Y1 = %d X2 = %d Y2 = %d \n ",upper_left_x,
upper_left_y, lower_right_x, lower_right_y );
   xmax_limit = lower_right_x - 64;
   ymax_limit = lower_right_y + 32;
   for (i_a= 0; i_a<= 512; i_a++)
      {
      data_array[i_a] = 0.0;
      }
   average = 0;
   average_2 = 0;
   st_dev = 0;
   npoint = 0;
   n_points = 0;
   n_points_off = 0;
   plot_peak = 0;
   peak_to_now = 0;
   rewind(file_pt2);
   for (scan_y = 0; scan_y <= ymax_limit; scan_y++)
      {
      fread (line_scan,sizeof(unsigned char),512,file_pt2);
      if( (scan_y>=(upper_left_y+32)) && (scan_y<=(lower_right_y+32)) )
         {
         for (scan_x=0; scan_x<=xmax_limit; scan_x++)
            {
            if( (scan_x>=(upper_left_x - 64)) &&
               (scan_x<=(lower_right_x - 64)) )
               {
               data_byte = line_scan[scan_x];
               data_float = data_byte;
               plot_peak = peak_detect(data_byte);
               average = average + data_float;
               average_2 = average_2 + (data_float * data_float);
               n_points = n_points + 1;
               data_array[npoint] = data_float;
               npoint = npoint + 1;
               }
            }
         }
      }
   average = average / n_points;
   average_2 = ( average_2 - n_points * average * average );
// Find the standard deviation //
   if( average_2 <= 0 )
      {
      st_dev = 0.0;
      }
   else
      {
      st_dev = sqrt( ( average_2 / n_points ) );
      }
// Find the correlation length //
// First repeat the array in data_array
```

```
   for (i_a= 0; i_a<= npoint-1; i_a++)
      {
      data_array[i_a] = data_array[i_a] - average;
      data_array[i_a + npoint] = data_array[i_a];
      }
   for (i_a= 0; i_a<= npoint; i_a++)
      {
      for (j_a= 0;j_a <= npoint; j_a++)
         {
         corrarray[i_a] = corrarray[i_a] +
         data_array[j_a] * data_array[j_a + i_a];
         }
      corrarray[i_a] = corrarray[i_a] / n_points;
      }
   for (j_a= 0;j_a <= npoint; j_a++)
      {
      f_data = corrarray[j_a];
      fprintf(file_pt3, "%f \n", f_data);
      }
   cor_value = corrarray[0] * 0.367879;  // e^-1
   i_a = 0;
   do   // while correlation value is greater than cor_value //
      {
      i_a = i_a + 1;
      }while( (corrarray[i_a] > cor_value)&&(i_a < npoint));
   index_cor = i_a;
// Print this information out //
   fprintf (stdprn,"      Number of Points = %6.0f \n ",n_points );
   fprintf (stdprn,"      Peak = %d \n ",plot_peak );
   fprintf (stdprn,"      Mean = %6.2f \n ",average );
   fprintf (stdprn,"      Standard Deviation = %6.2f \n ",st_dev );
   fprintf (stdprn,"      First Correlation Value = %6.2f \n ",corrarray[0] );
   fprintf (stdprn,"      Correlation Length Value = %6.2f \n ",
                          corrarray[index_cor] );
   fprintf (stdprn,"      Correlation Length in Pixels = %d\n ",index_cor);
   }
// ********************************************************************** //
void correl_line(void)
{
// Establish the cursor //
   Result  = gmouse.Mreset();
   setwritemode( COPY_PUT );
   gmouse.Set_Cursor( CIRCLE );
   gmouse.Mshow( FALSE );
   gmouse.Mshow( TRUE );
   gmouse.Mxlimit( 64, 628 );
   EXIT1 = FALSE;
   do                // while( EXIT1 != TRUE )
      {
      PositionR = gmouse.Mpressed( ButtonR );
// Cycle through selection until satisfied //
      if( TPos( PositionR.xaxis, 578, 628 ) )
         {
         if( TPos( PositionR.yaxis, 20, 40 ) )
            get_pt1();
         if( TPos( PositionR.yaxis, 50, 70 ) )
            get_pt2();
         if( TPos( PositionR.yaxis, 80, 100 ) )
            {
            EXIT1 = TRUE;
//                  Draw a box around the area //
```

```
            gmouse.Mshow( FALSE );
// Find the long dimension     //
            del_x = lower_right_x - upper_left_x + 1;
            del_y = lower_right_y - upper_left_y + 1;
            if (del_y >= del_x)
               {
               lower_right_x = upper_left_x;
               del_x = 1;
               }
            else
               {
               lower_right_y = upper_left_y;
               del_y = 1;
               }
            rectangle( upper_left_x, upper_left_y,
               lower_right_x, lower_right_y );
            }
         }
      } while( EXIT1 != TRUE );
   }
```