2

US ARMY
LABORATORY COMMAND
MATERIALS TECHNOLOGY LABORATORY

TECHNOLOGY TO WIN

MTL TR 88-34

INTERACTIVE ULTRASONIC NONDESTRUCTIVE EVALUATION

October 1988

C. H. CHEN and S. K. SIN
Information Research Laboratory, Inc.
415 Bradford Place
North Dartmouth, Massachusetts 02747

FINAL REPORT

Contract DAAL04-88-C-0003

Approved for public release; distribution unlimited.

Prepared for

U.S. ARMY MATERIALS TECHNOLOGY LABORATORY
Watertown, Massachusetts 02172-0001

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

Mention of any trade names or manufacturers in this report shall not be construed as advertising nor as an official indorsement or approval of such products or companies by the United States Government.

DISPOSITION INSTRUCTIONS

Destroy this report when it is no longer needed
Do not return it to the originator

Table of Contents

1. Introduction

It has been shown [9][10] that modern DSP (Digital Signal Processing) and Pattern Recognition techniques are very effective tools for Ultrasonic NDE (Non-Destructive Evaluations) of materials. In our recent researches [1], such techniques as High Resolution Spectral Analysis, Deconvolution, K-mean Clustering, Non-linear Mapping and Multipe Fisher's Discriminant, have been applied to the classification of hidden defect geometries with high accuracy.

In view of this uprising needs for DSP and Pattern Recognition in Ultrasonic NDE, a software package that effectively combines the two is in order. Interactive Ultrasonic NDE (IUNDE) TM developed by Information Research Laboratory, Inc. is such a package created especially to meet the needs of researches and practising engineers in this field. It also includes a variety of feature extraction functions that allow the user to choose features extracted from the time series of his or her interest for classification. The power of this package is further enhanced by its *versatile graphics* display capabilities, which can produce scatter plots, 2-dimensional and 3-dimensional graphics on a monitor and on a plotter or printer.

(JTS)

In fact, the DSP and Pattern Recognition libraries in this package is so rich that it can be used *either as a signal processing or a pattern* recognition package or both without limiting to NDE applications.

2. Software and Hardware Requirements

Software requirement :

1. MS DOS (R) 3.0 or above.
2. The NDE package is developed under the Microsoft (R) C 5.0 TM environment using the large memory model. It is strongly suggested that the same compiler (5.0 or above) be used in case the user needs to link this package with his own applications.

Hardware requirement :

1. IBM's PC/XT or PC/AT or compatibles.
2. Graphics adaptor (HGA, CGA and EGA) installed.
3. At least 280 Kbytes of free on-board memories.

3. Setting up the package

Before starting to run this package, the user should first set up his programming environment for graphics display by running the **EQUIP.exe** program. This program will create the file **CONFIG.gpc** which specifies the type of printer/plotter, monitor and graphics adaptor that the user is using.

> *NOTE : If you have a harddisk, the most convenient way is to copy all the files into the harddisk and set the harddisk drive as the default drive. Anyway, the files* **CONFIG.gpc, CONFIG.pen, SIMPLEX.fnt** *and* **COMPLEX.fnt** *must be in the default drive.*

4. Function Descriptions and Operation Guide

Besides the **EQUIP.exe** program which is used for package set-up, there are 3 executable files: **DEMO.exe**, **NDE.exe** and **WIG.exe**.

4.1 DEMO.exe

DEMO.exe is a demonstration of some of the functions offered by this package. It requires to read 3 data files: **SAMPLE**, **SA** and **SB.** It takes about 2 minutes' time to read all these data on an IBM-PC/XT, so " **Be Patient !** "

4.2 NDE.exe

NDE.exe can perform 15 kinds of digital signal processing or pattern recognition operations. The available options are tabulated in the **MAIN_MENU**.

MAIN_MENU

bsp	Power Spectrum by Burg's Technique
near	Nearest Neighbour Classification
cr	Cross-Correlation between two waveforms
dis	Graphics Display
fe	Feature(s) Computation
fsp	Magnitude Spectrum by FFT
hb	Find Analytic Part of the input signal by FFT
nlmp	Nonlinear Mapping
spi	Spiking Filter Deconvolution (no IBM-PC version)
wv	Wavelet Transformation
win	Wiener Filtering Deconvolution using FFT
wind	Hann's frequency windowing (a low-pass filter)
fish	Multiple Fisher's Discriminant
foly	Foley-Sammon Transformation
kmean	K-mean Clustering

Once entered the main_menu, the user is prompted to provide his choice. The user can choose the desired operation by entering the corresponding mnemonics in small letters (e.g. **bsp** <return>).

After entering an option, the user should be ready to answer a few questions to provide sufficient informations that are necessary for that particular operation (e.g., where to get the input data; how long is the input data; where to store the results; etc.). In the following few sections, we are going to discuss each of these operations in some details. In most of the cases, we assume the input data to be *read from file(s) and the results* to be stored in file(s). The user will be prompted to provide the necessary filenames in these cases.

In the following subsections, we will describe each of these functions in some detail. Unless stated otherwise, $x(n)$ and $x_0(n)$ are used to denote the Test Signal and the Reference Signal respectively while $X(k)$ and $X_0(k)$ are used to denote their Fourier Transforms.

4.2.1 bsp (fig.3)

description

This function computes the power spectral density of the input signal by Burg's technique [2]. This is a model-based spectral estimation technique. The input time series is modeled as an autoregressive(AR) process driven by white noise, such that

$$X(k) = 1 \Big/ \Big(1 + \sum_{i=1}^{p} a[i]\exp(-j2\pi ik/N) \Big)$$

where P is known as the AR order and the a[i] 's are known as the AR coefficients. There are no general rules for choosing an optimal value of P. As a rule of thumb, this can be taken to be about one-third of the record length N for small N.

The computation is divided into 2 steps. First we compute the a[i]'s using Burg's Technique which is a very popular one among some other possible methods. Then we compute $|X(k)|^2$ using FFT.

informations required

1.1 Filename of the input data (e.g. T15a0.col).

1.2 Starting point and ending point of data (e.g. 100,355).

1.3 Length of the input data (e.g. 256).

note: maximum is 512

2.1 Desired order of the AR model (e.g. 20).

2.2 Number of spectral points required (e.g. 200).

note: This is a resolution issue. The user should be aware to choose a sufficiently large number such that no significant structures of the power spectrum will be missed. The maximum allowed is 512.

3.1 If you want to store the AR coefficients, you need to provide a filename to store these values. The number of output points to be stored in this example should be 20 (equals to the AR order used).

4.1 An output filename to store the power spectrum (e.g. psd.dat).

4.2 Number of output points to be stored (100 for this example). This is because the input signal is real which implies a symmetrical power spectrum.

4.2.2. near

description

Suppose we have *c* classes of *n*-vectors in an *n*-dimensional feature space and that each class contains *P* patterns. Let **t** = $(t_1,t_2,...,t_n)$ be the test vector whose class identity is of interest and **p** = $(p_1,p_2,...,p_n)$ be one of the known-class patterns (or training pattterns). The Euclidean distance

$$d = \sqrt{\left\{ \sum_i (t_i - p_i)^2 \right\}}$$

is computed for all **p** . Then, by the nearest neighbor rule, the class of **t** is taken as the same as that of the **p** with the smallest d.

informations required

1.1 Number of features in each pattern (e.g. 3).

1.2 Total number of training patterns (e.g. 12).

1.3 A filename for each feature (i.e., 3 filenames needed in this example).

1.4 Starting points and ending points to read these features from each of the files.

1.5 Number of clusters in the training-set (e.g. 4).

note: For this example, let the 3 inputfiles be File_1, File_2 and File_3. Therefore, File_1, File_2 and File_3 should contain respectively the feature(1)'s, feature(2)'s and feature(3)'s of all training-patterns. In other words, 12 pieces of data are to be read from each of these files in this example. The relative order of these patterns' features in each file must be the same and that we have assumed that there are equal number of training patterns in each cluster (3 training patterns per cluster in this example).

4.2.3. cr (fig.5)

description

This function computes the normalized cross-correlation of 2 input signals. If we let x(k) be the first signal and y(k) be the second signal, this subroutine will compute:

$$r_{xy}(m)/\{ r_{xx}(0)r_{yy}(0) \}^{1/2}$$

where $r_{xy}(m) = \sum_{k} x(m+k)y(k)$

informations required

1.1 Filename of the first data set.

1.2 Starting point and ending point of the data.

1.3 Filename of the second data set.

1.4 Starting point and ending point of the data.

1.5 Length of each data set (maximum is 256).

2.1 An output filename to store the result of cross-correlation.

2.3 Number of output points need to be stored. This would be equal to 2 times minus 1 the number given in 1.5 above.

4.2.4. dis

description

Graphics display.

available options

(1) scatter plot, or (2) waveform plot.

informations required

(i) For scatter plots:

1.1 Number of clusters to be plotted.
1.2 Number of points on each cluster.
1.3 Choose a symbol to represent each cluster on the plot.

note: a symbol is an integer from 0 - 7. These symbols are respectively shown as follows:

□ ○ ▽ + X ◇ △ ■

2.1 A filename to read the data to be plotted. Each piece of data should contain 2 parts - the (x,y) coordinates.
3.1 The title for the plot.
3.2 The label on the x-axis.
3.3 The label on the y-axis.

(ii) For waveform plots

1.1 Filename of the input waveform data.
1.2 Starting point and ending point of the data to be plotted.
2.1 The title for the plot.
2.2 The label on the x-axis.
2.3 The label on the y-axis.
2.4 Number of data points to be plotted.

At the end of each plot, the user can return to the **main menu** by hitting the **<return>** key or he can dump the plot onto a printer/plotter by hitting the **<m>** key and specifying the location and size of the plot on the paper. However, the user may suffer from a shortage of system memory that forbids him to dump the plot onto the printer/plotter. If this is the case, the user may want to use a **DOS** command to dump the on-screen plot onto a printer (or he can use some other package that will do the job).

4.2.5. fe

description

This is a feature extraction subroutine. The several options included here have been tested to be very effective in ultrasonic NDE problems .

available options

ar	Amplitude Ratio.
bk	Store the computed features and return to the main menu.
bnw	Bandwidth of the input spectrum.
kusk	Kurtosis and Skewness of the input time-series.
fr	Frequency Ratio.
mcr	Maximum of Cross-correlation with reference.
fkusk	Kurtosis and Skewness of the input spectrum.
sft	Show the computed features in tabulated form.

After entering an operation (in small letter), again, the user is prompted to answer a number of questions to supply sufficient informations that are necessary for that particular operation. These operations are explained below.

(i) ar

description

Amplitude ratio (ar) is defined as:

$$ar = \left\{ \sum |x(k)| \right\} / \left\{ \sum |x_0(k)| \right\}$$

informations required

1.1 Number of data points in each of the input signals.

1.2 Filenames, starting points and ending points of the referenced signal and the test signal.

(ii) **bk**

description

This option allows the user to store the computed features in a file and returns him to the main menu.

informations required

If the user wants to store the computed features, he will be prompted to provide an ouput filename.

(iii) **bnw**

description

This option computes the bandwidth of the test signal from its magnitude spectrum. It also computes the frequency of peak magnitude as well as the fractional power distribution in 10 equally spaced frequency bands (starting from d.c. to the 100th frequency bin).

note: The input to this operation should be the magnitude spectrum.

information required

1.1 Number of spectral data points available in the input magnitude spectrum.

1.2 Filename of the input spectral data.

(iv) **kusk**

description

This function computes the kurtosis and skewness of the input signal, say x(k), where:

$$\text{Kurtosis} = \left\{ \sum_k (k - \mu)^4 x(k) \right\} \Big/ \left\{ \sum_k (k - \mu)^2 x(k) \right\}^2$$

$$\text{Skewness} = \left\{ \sum_k (k - \mu)^3 x(k) \right\} \Big/ \left\{ \sum_k (k - \mu)^2 x(k) \right\}^{3/2}$$

$$\mu = \sum_k k \cdot x(k) \Big/ \sum_k x(k)$$

(v) **fr**

description

The definition of Frequency Ratio (*fr*) is the same as that of *ar* in (i) above except that $|x(t)|$ and $|x_0(t)|$ are replaced by $|X(f)|$ and $|X_0(f)|$ respectively.

informations required

Same as (i) except that the input data $x(t)$ and $x_0(t)$ are replaced by $|X(f)|$ and $|X_0(f)|$ respectively.

(vi) **mcr**

description

Let $a(k)$ be a referenced signal and $x(k)$ the test signal. Define cross-correlation $r_{ax}(m)$ as :

$$r_{ax}(m) = \sum_k a(m+k)x(k)$$

mcr computes : $\mathrm{Max}\left\{ r_{ax}(m)/\sqrt{(r_{aa}(0)r_{xx}(0))} \right\}$
as well as the value of m at which this maximum occurs.

informations required

1.1 Number of data points in each set of data.

1.2 Filenames, starting points and ending points of the referenced data and the test data.

(vii) **fkusk**

description

This function computes the Kurtosis and Skewness (see (iv) above for definitions) of a power spectrum. The input to this function should be a power spectrum.

informations required

Same as (iv) except that $x(t)$ is replaced by $|X(f)|^2$ as the input.

(viii) **sft**

description

This option allows the user to view all the computed features on the screen in a tabulated form.

informations required

None.

4.2.6. fsp (fig.4)

description

This function computes the power spectrum of the input signal using the FFT algorithm.

informations required

1.1 Filename of the input data.

1.2 Starting point and ending point of the data.

2.1 Number of spectral points required (power of 2, maximum is 512).

2.2 FFT order (must be consistent with the answer to 2.1, e.g., if you enter 256 in 2.1, you should enter 8 here; maximum is 9)

3.1 An output filename to store the computed magnitude spectrum.

3.2 Number of output points to be stored.

4.2.7. hb (fig.10)

hb stands for Hilbert Transform; the real part and imaginary part of an analytic signal are Hilbert transforms of each other.

description

This function computes the analytic signal corresponding to the input using signal FFT.

Let s(t) be a complex signal such that

$$s(t) = s_r(t) + j\, s_i(t)$$

where $s_r(t)$ and $s_i(t)$ are both real functions.

If $s(t)$ is an analytic signal, *then* $s_i(t)$ *is the Hilbert transform* of $s_r(t)$ or $s_r(t)$ is the negative Hilbert transform of $s_i(t)$. Another important property of the analytic signal $s(t)$ is that $S(f) = 0$ for all $f < 0$.

Our implementation of this function is based on this latter property. First we compute X(k) using FFT. Then we set X(k) = 0 for all negative frequencies and the analytic signal is then obtained by inverse FFT. The magnitude of the resulting analytic signal is then computed and stored. This signal will exhibit the energy distribution of the input in the time domain.

informations required

1.1 Filename of the input signal.
1.2 Starting point and ending point of the signal.
2.1 Number of points of analytic signal required (power of 2, maximum is 512).
2.2 FFT order (refer to (f)-2.2).
3.1 An output filename to store the computed analytic signal.
3.2 Number of output points need to be stored.

4.2.8. nlmp

description

Nonlinear Mapping is an algorithm that maps vectors from, say, an L-dimensional space to some lower dimensional space (for our case, this is usually a 2 or 3 dimensional space).

Let $\mathbf{X}_l$ be vectors in the origina[illegible] .nd $\mathbf{Y}_l$ be the corresponding vectors (i.e. after mapping $\mathbf{X}_l$) in the lower dimensional space. Let d_{ij}^* denotes the Euclidean distance between $\mathbf{X}_l$ and $\mathbf{X}_j$ and d_{ij} the Euclidean distance between $\mathbf{Y}_l$ and $\mathbf{Y}_j$. This algorithm then seeks the optimal $\mathbf{Y}_j$'s in the sense that

$$E = (1/c)\sum_{i<j}(d_{ij}^* - d_{ij})/d_{ij}^*$$

where

$$c = \sum_{i<j} d_{ij}^{*} \quad ,$$

is minimized.

The details of the derivation of this algorithm is described in [3]. This is an iterative algorithm such that iteration stops when E is less than some prescribed error bound supplied by the user.

informations required

1.1 Dimension of the original space (say m).
1.2 Number of m-vectors to be mapped.
1.3 Desired dimension of the space to be mapped onto.
1.4 A filename to read the input m-vectors (m components in each input vector).
2.1 Since an iterative algorithm is used in this part. Iteration will stop when either the number of iterations reaches 1000 or the error measure becomes less than some error bound specified by the user. At this point, the user will be prompted to provide an errror bound.
3.1 After a successful run, the user will be prompted to provide an output filename to store the lower dimensional vectors.

4.2.9. **spi** (fig.13-16)

description

Spiking filtering [5] is a deconvolution operation by which we estimate the transfer function of a defect (which we model as a linear time-invariant system) with respect to some reference.

The first step is to find a spiking filter h(k) which reduces the reference signal (widely known as a wavelet in reflection seismology) to $\partial(k-k_0)$, which is a time-shifted unit sample, i.e.

$$h(k) * x_0(k) = \partial(k-k_0)$$

where * denotes the convolution operator and k_0 is a variable to be determined.

$$\text{Let} \quad e = 1/R = \sum_k \left(h\hat{}(k) * x_o(k) - \partial(k-k_o) \right)^2$$

where h^(k) is a least mean square error estimate of h(k). Then k_o is chosen as such that e is minimum or R is maximum. R is known as the Performance Index.

Once an optimal h^(k) (in the least mean square error sense) has been obtained, the transfer function is estimated, up to a linear time shift ko, by h^(k)$*x_o$(k).

informations required

1.1 Filename, starting point and ending point of the reference signal.
1.2 Filename, starting point and ending point of the test signal.
1.3 A filename to store the deconvolved signal.

note: spi is not available on the IBM-PC *version of this package. Some results can be found in the* DEMO.exe *program.*

4.2.10. wv (fig.6)

description

Let x(k) be the input signal, $0 \leq k \leq N-1$. The wavelet transformation $X(\mu,\beta)$ is defined as:

$$X(\mu,\beta) = (1/\sqrt{\mu})\sum_k \left\{ x(k)\cos((k-\beta)/a)\exp(-((k-\beta)/\mu)^2/2) \right\}$$

$$\text{for} \quad 0 \leq \beta \leq N-1 \quad ;$$

where μ is called a scale factor.

wv will compute $X(\mu,\beta)$ of up to 8 different values of μ and display all transformed wavelets on one single plot.

From our experience, although the pulse echoes from hidden defects of different geometries may themselves look very similar, the wavelet transform patterns they produced are often quite different.

informations required

1.1 Filename, starting point and ending point of the input signal.

1.2 Number of data points in the input wavelet.

1.3 Number of differnt scales required; the first scale and the step size between scales. All these must be integers.

4.2.11. win

description

This is another deconvolution operation using the method of Wiener Filtering [5].

The Wiener Filter is given by

$$H_w(k) = X_o^*(k)/\{ |Xo(k)|^2 + 1/SNR(k) \}$$

where $SNR(k) = Sx_o(k)/Sn_o(k)$

is the ratio of the power spectral density of $x_o(k)$ to that of the noise.

To avoid the estimation of these power spectral densities, SNR(k) is usually replaced by a small positive constant 1/q, where q is called the noise desensitizing factor. As rule of thumb, q is taken to be about 0.01 times the peak value of $|X_o(k)|^2$. The required transfer function is then estimated by

$$\text{Inverse FFT}(H_w(k)X(k))$$

informations required

1.1 Filename, starting point and ending point of the referenced signal.

1.2 Filename, starting point and ending point of the test signal.

2.1 Provide a noise desensitizing factor q. As a rule of thumb, q can be chosen to be about 0.01 times the maximum of the magnitude spectrum of the referenced signal.

3.1 Number of spectral points to be used in the filtering process (power of 2, maximum is 512).

3.2 FFT order (refer to (f)-2.2).

4.2.12. wind (fig.11-12)

description

This is a 3-point moving average operation which is equivalent to a Hann's frequency window. In particular, if x(k) is the input signal, then x(k) will be replaced at the output by

$$0.25\{ x(k+1) + x(k-1) \} + 0.5x(k).$$

required informations

1.1 Filename, starting point and ending point of the input data to be filtered.

1.2 Total number of points to be filtered.

2.1 An output filename to store the filtered signal.

2.2 Number of output points need to be stored.

4.2.13. fish (fig.22)

description

Multiple Fisher's Discriminant is a feature space compression algorithm. The details for its derivation can be found in [6].

Suppose at the beginning, we have c classes (say c_1,c_2,.... and c_c) each of P patterns in an n-dimensional feature space, where $n \geq c$. This operation can optimally map these n-vectors into ,for instance, an m-dimensional space such that $m \leq c-1$.

Let **x** be one of the available patterns of known class in the starting feature space. We need to find an optimal linear transformation, an n by m matrix **W** to map each vector **x** into a vector **y** in the lower dimensional space such that

$$y = W^T x$$

First, we compute the within class and between class scatter matrices which are denoted respectively by S_w and S_b. These matrices are defined as follows :

$$S_w = \sum_{i=1}^{c} \sum_{\text{all } x \text{ in } C_i} (x-m_i)(x-m_i)^T$$

$$S_b = \sum_{i=1}^{c} P(m_i-m)(m_i-m)^T$$

where

$$m_i = (1/P) \sum_{\text{all } x \text{ in } C_i} x$$

$$m = (1/cP) \sum_{i=1}^{c} Pm_i$$

The matrix W is found such that the criterion function

$$J(W) = \left|W^T S_b W\right| / \left|W^T S_w W\right|$$

is maximized.

The solution to this problem can be obtained by solving

$$S_b w_i = \mu_i S_w w_i$$

where μ_i is known as the generalized eigenvalues and $\mathbf{w}_i$ the corresponding generalized eigenvectors.

It can be easily shown that the rank of $\mathbf{S}_b$ is no greater than c-1 which implies that no more than c-1 of the eigenvalues are non-zero. Now, let m be the number of non-zero eigenvalues and define

$$W = \{ \mathbf{w}_1 \;\; \mathbf{w}_2 \dots \;\mathbf{w}_m \}$$

where $\mathbf{w}_1$, $\mathbf{w}_2$, ... $\mathbf{w}_m$ are the eigenvectors that correspond to those non-zero eigenvalues. These vectors are also referred to as the direction vectors for obvious reasons.

Due the finite precision of our computers, the "unused" eigenvalues will not be exactly zero. Therefore, in practical implementation, we need to monitor the relative sizes of all eigenvalues and determine which eigenvalues should be considered as zero. (From our experiences, those eigenvalues that is less than 0.01 times the maximum eigenvalue can safely be taken as zero's.)

informations required

1.1 Number of clusters of training patterns (maximum is 5).
1.2 Number of features per pattern (maximum is 6).
1.3 Number of training patterns per cluster (maximum is 54).
1.4 A filename for each cluster of training patterns.

note: We have assumed that each cluster's pattern data are stored in a different file. For example, if you have 3 clusters, each of 10 training patterns and each pattern consists of 5 features, you need to store these data in 3 files, and each file contains 50 pieces of data.

After entering the above informations, the following informations will be dumped on the screen:

(i) The **between-class scatter matrix** ($\mathbf{S}_b$) .
(ii) The **within-class scatter matrix** ($\mathbf{S}_w$) .
(iii) The generalized eigenvalues and generalized eigenvectors of ($\mathbf{S}_b$, $\mathbf{S}_w$).

2.1 At this point, the user is prompted to provide a threshold to choose eigenvalues. Usually, this threshold is less than

0.01 times the maximum eigenvalue. Eigenvalues less than the threshold will be deleted in the subsequent computations.

3.1 If no numerical ill-condition is encountered, the user will be prompted to provide an output filename to store the results at the end of the computations.

4.2.14. foly (fig.21)

description

Foley Sammon Transformation [8] can be used for feature space compression when there are only two classes in the feature space of interest.

Suppose we have 2 classes of patterns, say c_1 and $c_{2,}$ in an n-dimensional feature space (the starting space) and we want to map these patterns into an m-dimensional space (the ending sace) with $m \leq n$. Let **x** be a pattern (or vector) in the starting space and **y** its correspondence in the ending space. We need to find m n-vectors , say $\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_m$ and that

$$\mathbf{y} = \mathbf{W}^T\mathbf{x} \quad \text{where} \quad \mathbf{W} = \{ \mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_m \}$$

These vectors are called direction vectors.

Define the criterion function

$$J(\mathbf{w}) = \left(\mathbf{w}^T(\mathbf{m}_1 - \mathbf{m}_2)\right)^2 / \mathbf{w}^T S_w \mathbf{w}$$

where $\mathbf{m}_1$ and $\mathbf{m}_2$ are the mean vectors of c_1 and c_2 respectively and $\mathbf{S}_w$ is the within class scatter matrix (see 4.2.13). This is the the same criterion function used in Fisher's linear discriminant.

The first direction vector $\mathbf{w}_1$ is computed as such that $J(\mathbf{w}_1)$ is maximized. Since our primary concern is the direction of $\mathbf{w}_1$, we may impose an additional condition that $\mathbf{w}_1^T\mathbf{w}_1 = 1$. The solution is found as

$$\mathbf{w}_1 = \beta_1 \mathbf{S}_w^{-1}\mathbf{d}$$

$$\beta_1{}^2 = \mathbf{d}^T(\mathbf{S}_w^{-1})^2\mathbf{d}$$

where $\mathbf{d} = (\mathbf{m}_1 - \mathbf{m}_2)$

Then $\mathbf{w}_2$ is found also by maximizing $J(\mathbf{w}_2)$ with the additional constraints of $\mathbf{w}_2^T\mathbf{w}_2 = 1$ and $\mathbf{w}_2^T\mathbf{w}_1 = 0$. The process is carried on until $\mathbf{w}_m$ is obtained by maximizing $J(\mathbf{w}_m)$ with additional constraints that $\mathbf{w}_m^T\mathbf{w}_m = 1$ and that $\mathbf{w}_m^T\mathbf{w}_i = 0$ for all $i \neq m$.

If we define the "discrim values" $r_i = J(\mathbf{w}_i)$, we shall observe that

$$r_1 \geq r_2 \geq .... \geq r_m$$

informations required

note: The user should first provide 2 clusters of training patterns. We have assumed that each cluster contains the same number of patterns and that each cluster's data are stored in a separate file, i.e. 2 input files are required.

1.1 Number of training patterns per cluster.
1.2 Number of features per pattern.
1.3 Desirable number of **compressed features** per pattern.
1.4 A filename to read the data of the first cluster.
1.5 A filename to read the data of the second cluster.

After these informations are entered, the following results will be dumped on the screen:

(i) The within-class scatter matrix $\mathbf{S}_W$.
(ii) The direction vectors $\mathbf{w}_i$.

2.1 Number of test patterns (whose festures are to be compressed).
2.2 A filename to read all these test patterns.
3.1 After a successful computation, the user will be prompted to provide a filename to stored the compressed features.

4.2.15. kmean

description

Kmean is a clustering algorithm which iteratively updates the elements of each cluster. Suppose there are P test patterns needed to be grouped into K clusters. We first make an initial guess of the K cluster centers (each cluster center is the mean vector of all the patterns in that cluster). If no a priori knowledge is available, these initial centers may be chosen arbitrarily (for instance, the first K of the P input patterns). Each pattern is then assigned to a cluster such that the Euclidean distance between this vector and the center of that cluster is the smallest. When all patterns have been assigned, each cluster center is updated by replacing it with the mean vector of the cluster. Iteration goes on until all cluster centers stablize.

informations required

1.1 Number of test patterns to be classified.
1.2 Number of features in each pattern.
1.3 A filename to read all these test patterns.
2.1 Desired number of clusters (K).
2.2 Guess the positions of the initial centers of each cluster in the input file.

After these informations are entered, conclusions will be dumped on the screen.

note: These results will not be stored in a file. The user may want to dump these on-screen results onto a printer using a DOS command.

4.3 Wig.exe

WIG.exe is another executable file that computes the **Wigner-Ville Distribution** [11] of an input signal and will provide a 3-D graphics display of the results. Two graphics at different view point will be displayed (one after the other).

description

Let x(t) be the input signal, the Wigner-Ville distribution is defined as:

$$W_x(t,\Omega) = \int x(t+\beta/2)\, x^*(t-\beta/2)\, \exp(-j\Omega t)\, d\beta$$

where $x^*(t)$ denotes the complex conjugate of x(t).

This is a time-frequency distribution and is believed to be capable of providing time-domain and frequency-domain informations simultaneously. Such a property is very useful for defect classfication problems since both time-domain and frequency-domain both provide relevant informations .

informations required

1. A filename to read the input signal.
2. Starting point (ist) and Ending point (ien) of this signal.
   note: Only a maximum of 64 points are allowed in the input data. Hence, we require that : **(ien - ist) < 64** .
3. Number of time points for which Wigner-Ville distribution is to be computed (maximum is 64).

5. Computer Results

In this section, we present some results that are obtained through the use of this package.

The reference signal r(t) and the test signal x(t) shown in figs.(1) and (2) are pulse echoes obtained by using a 15 MHz *ultrasonic transducer* and digitized with a sampling rate of 100 MHz. Both signals have a record length of 512. r(t) is the pulse echo from a flawless aluminium block while x(t) is the pulse echo from a block with a hidden angular cut.

Fig.(3) shows the Power Spectrum of r(t) computed by a 512-point FFT (Fast Fourier Transform). The frequency resolution is 512 points per 100 MHz.

Fig.(4) is the cross-correlation function $R_{rx}(\mu)$ for μ = -1500 ns to 1500 ns (note: this function has been shifted by 1500 ns so that the time origin is at zero).

Fig.(5) shows the (AR) Power Spectrum of r(t) computed by Burg's Technique. The AR order is 50 and the frequency resolution is 512 points per 100 MHz.

The *signal* $h_x(t)$ *shown is fig.*(6) is the impulse response extracted from x(t) by Wiener filtering with r(t) as the reference signal. The noise desensitizing factor used is equal to 0.01 times the peak power shown in fig.(3).

Fig.(7) is the result of time-averaging $h_x(t)$.

Fig.(8) shows the analytic signal of $h_x(t)$.

In fig.(9), we have simulated a pulse echo from a 2 layer composite material by concatenating 2 single reflections. We call this signal the composite signal c(t). The impulse response $h_c(t)$ extracted from this signal by Wiener Filtering (with r(t) as reference) is shown in fig.(10). Fig.(11) is a time-averaged version of $h_c(t)$ while fig.(12) is the analytic signal of $h_c(t)$.

In figs.(13) to (17) we demonstrate the different steps of a spiking filtering operation (note: the intermediate results shown here are not all accessible by the user of this package). First, let the reference signal (or the reference wavelet) be ref(t) as shown in fig.(13). The spiking filter (or inverse filter) $h_s(t)$ shown in fig.(14) is computed such that the performance index is maximum in the performance index profile shown in fig.(15). We may check the accuracy of this spiking filter by convolving ref(t) with $h_s(t)$ and the result of this convolution is shown in fig.(16). Now, let the test signal of interest be s(t) which is made up of 2 overlapped wavelets, such that $s(t) = ref(t) + 0.5ref(t-5)$. The final result, which is given by $s(t)*h_s(t)$, is shown in fig.(17). You may observe that the occurence of the two spikes in fig.(17) corresponds exactly to the occurence of the two wavelets in s(t). Fig.(18) is another example of spiking filtering in which the test signal is T15A1 and the reference signal is T15A0 ; these waveforms can be found in Appendix B of this report.

Figs.(19) and (20) show the Wigner-Ville distribution of x(t). The numbers of frequency points and time points are both equal to 64.

Fig.(21) and (22) are scatter plots which show the results of feature space compression from a 4-dimensional to a 2-dimensional space by Foley-Sammon Transformation and Multiple Fisher Discriminant respectively. In these figures, f1 and f2 denote the 2 dimensions after the compression. The original data used are Iris data which are tabulated in Table 1 on page 32 of this report. The 2 classes of patterns (Iris) we chose for this demonstration are "versicolor" and "setosa"; 50 patterns from each class have been used.

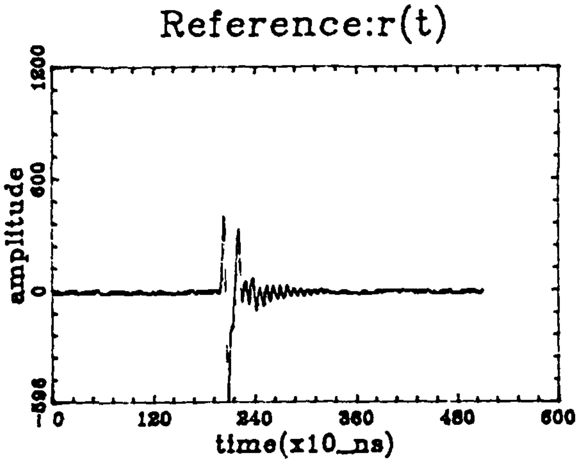


Fig. 1 A reference signal r(t)

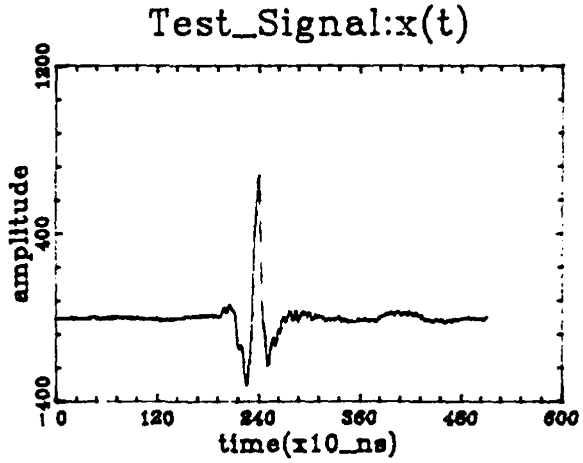


Fig. 2 A test signal x(t)

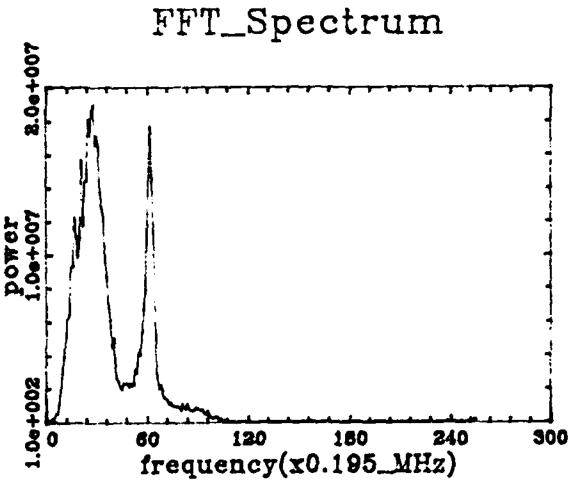


Fig. 3 Power spectrum of r(t) by FFT

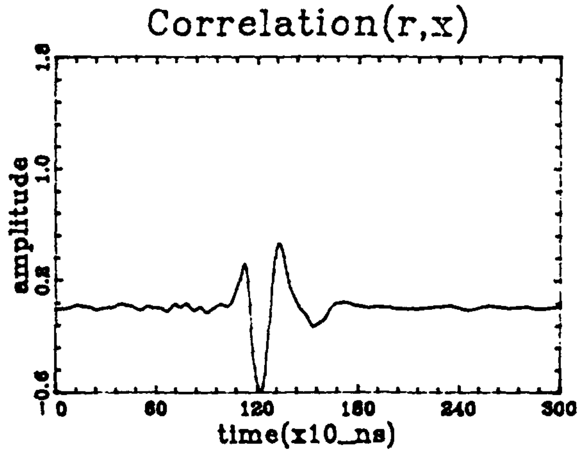


Fig. 4 Correlation function R_{rx}

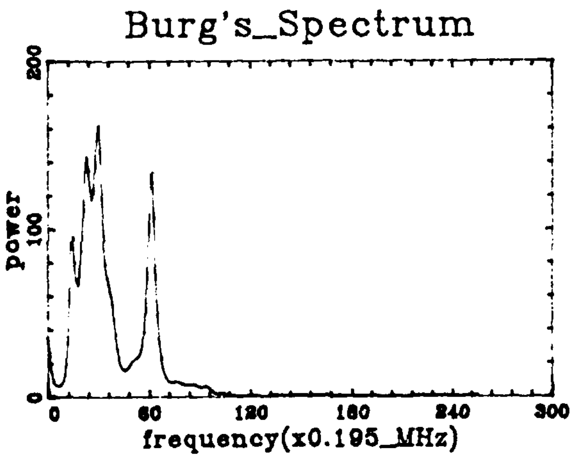


Fig. 5 Power spectrum of r(t) by Burg's technique.

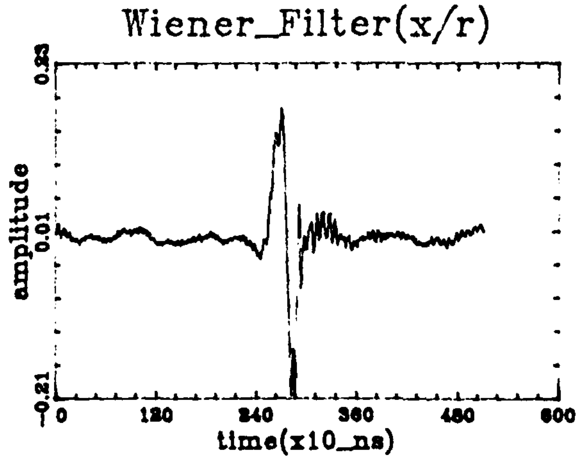


Fig. 6 $h_x(t)$, impulse response by Wiener filtering.

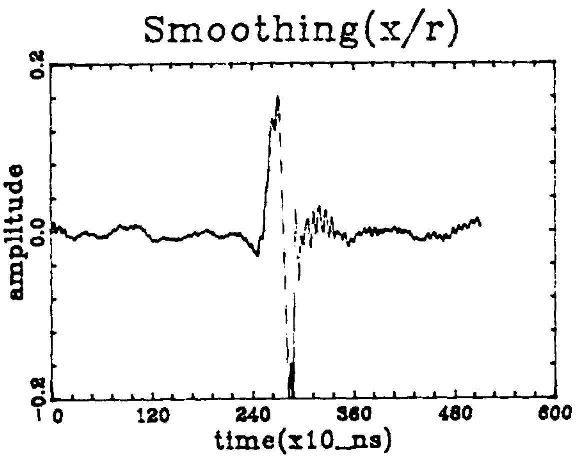


Fig. 7 On smoothing $h_x(t)$

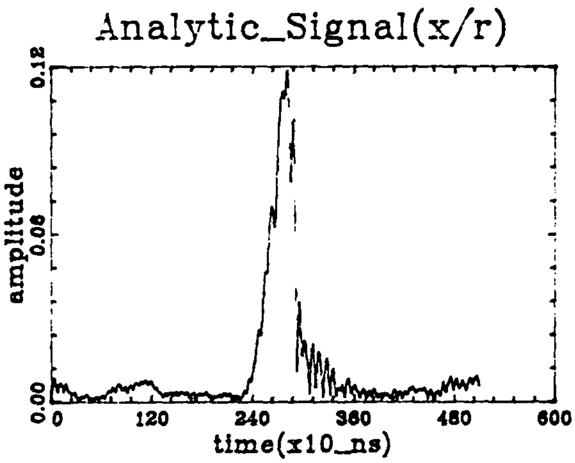


Fig. 8 Analytic signal of $h_x(t)$

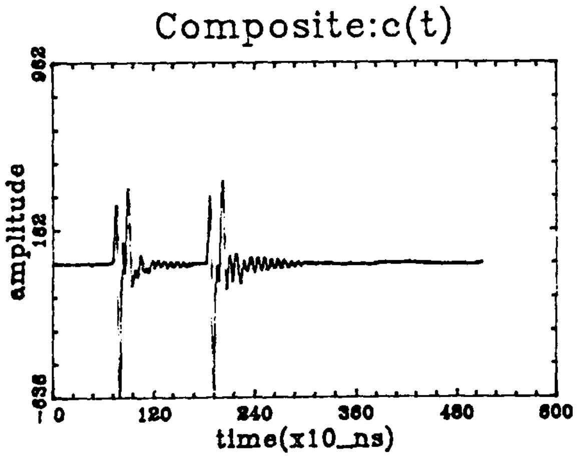


Fig. 9 A composite signal c(t)

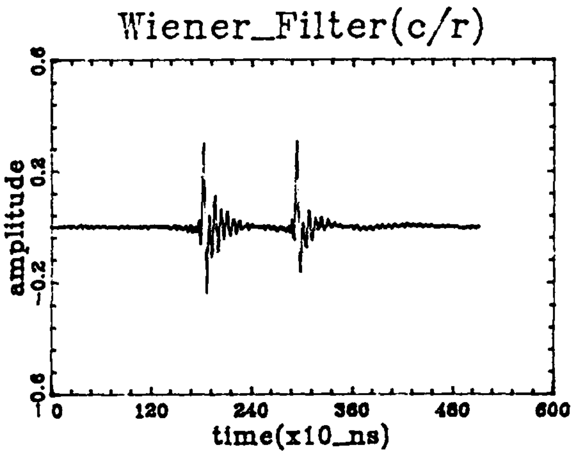


Fig. 10 $h_c(t)$, impulse response by Wiener filtering

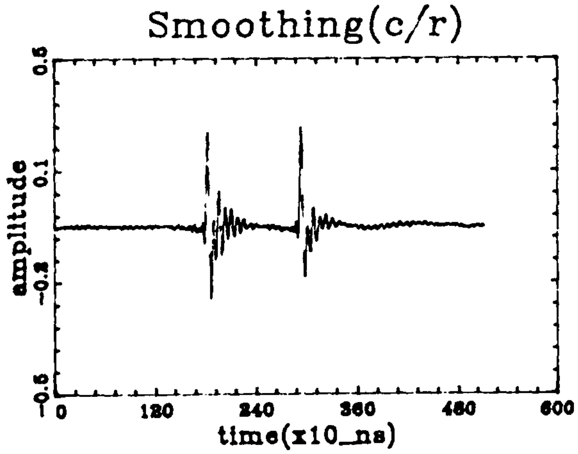


Fig. 11 On smoothing $h_c(t)$

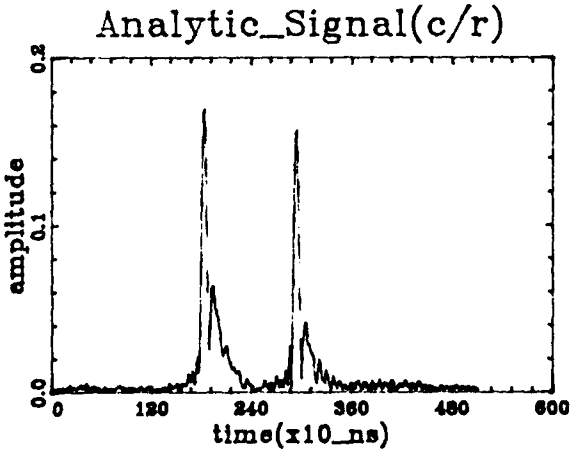


Fig. 12 Analytic signal of $h_c(t)$.

Figures 13-17 show steps of spiking filtering.

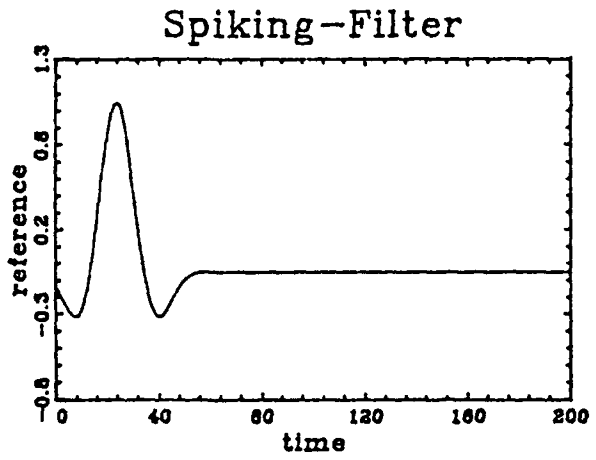


Figure 13 Reference waveform ref(t)

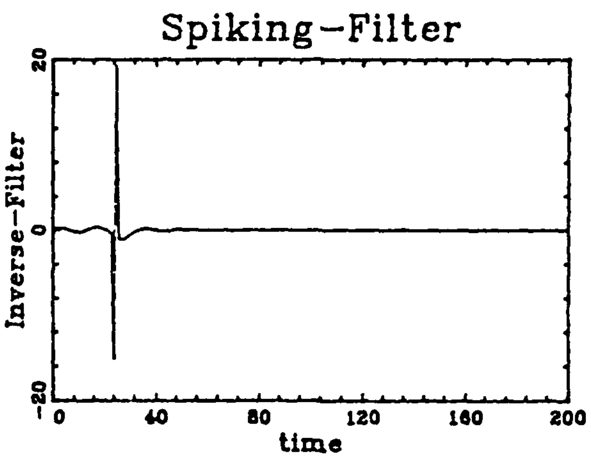


Figure 14 Inverse filter $h_S(t)$

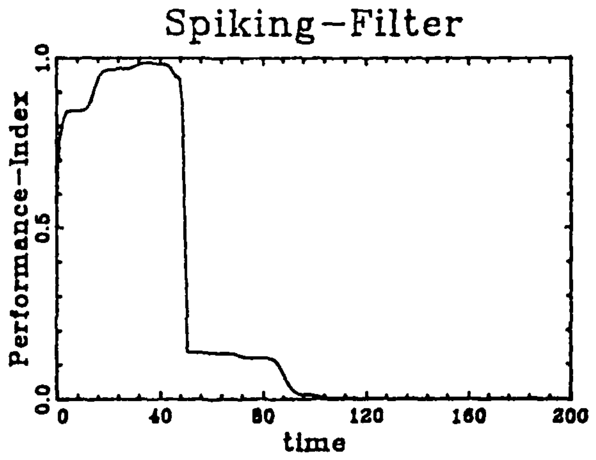


Figure 15 Performance measure $R(k_o)$

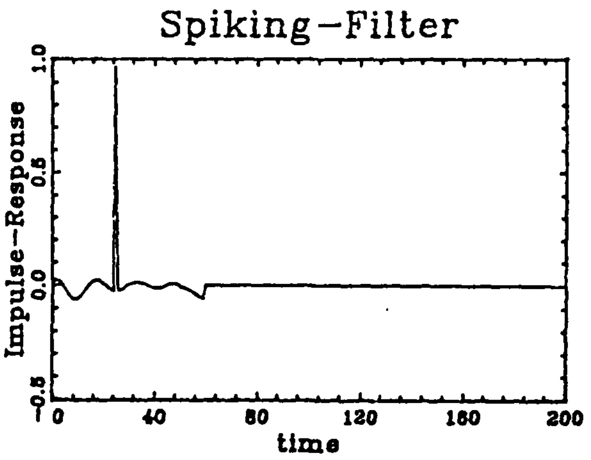


Figure 16 Result of spiking filtering ref(t)*$h_S(t)$

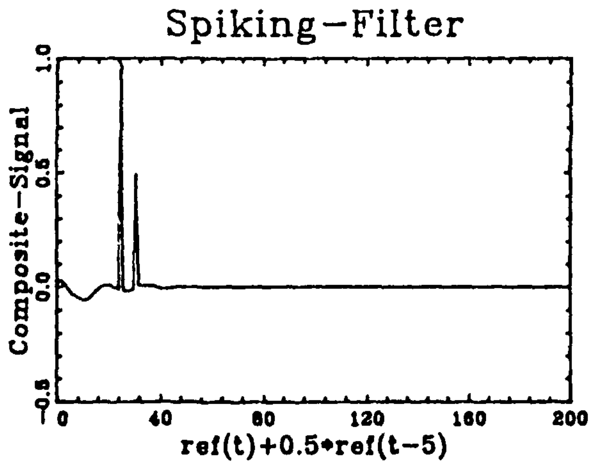


Figure 17 Result of spiking filtering composite signal, i.e. s(t)*$h_S(t)$, where s(t)= ref(t)+0.5ref(t-5)

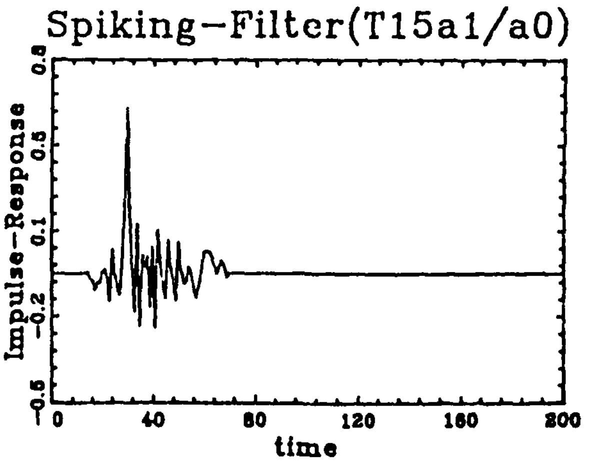


Figure 18 Result of spiking filtering T15 using T15A0 as reference.

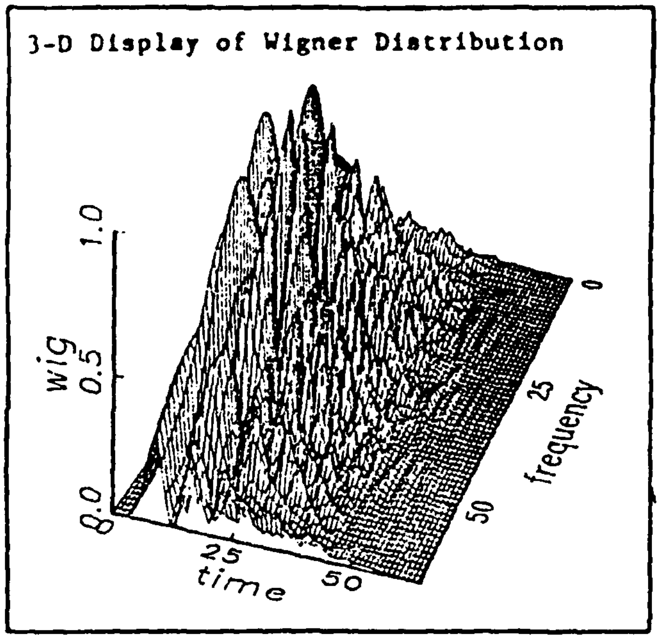


Fig. 19

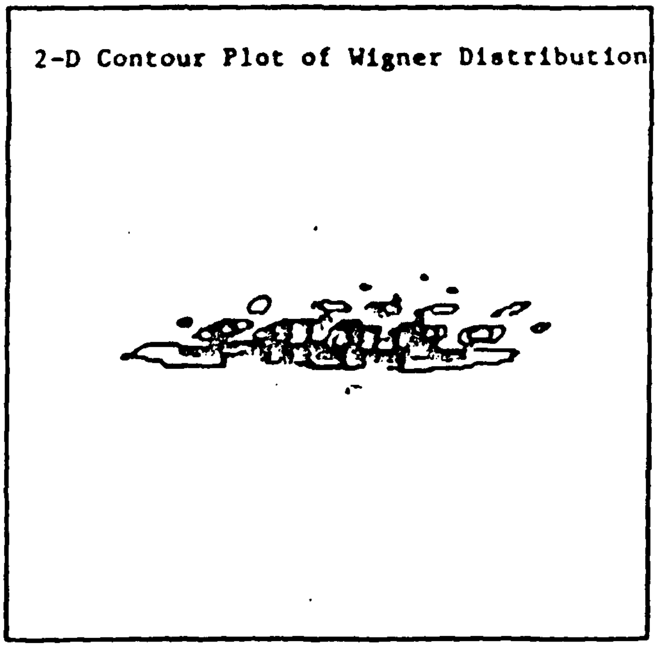


Fig. 20

Wigner distribution in 3-D display (Fig. 19) and 2-D contour plot (Fig. 20).

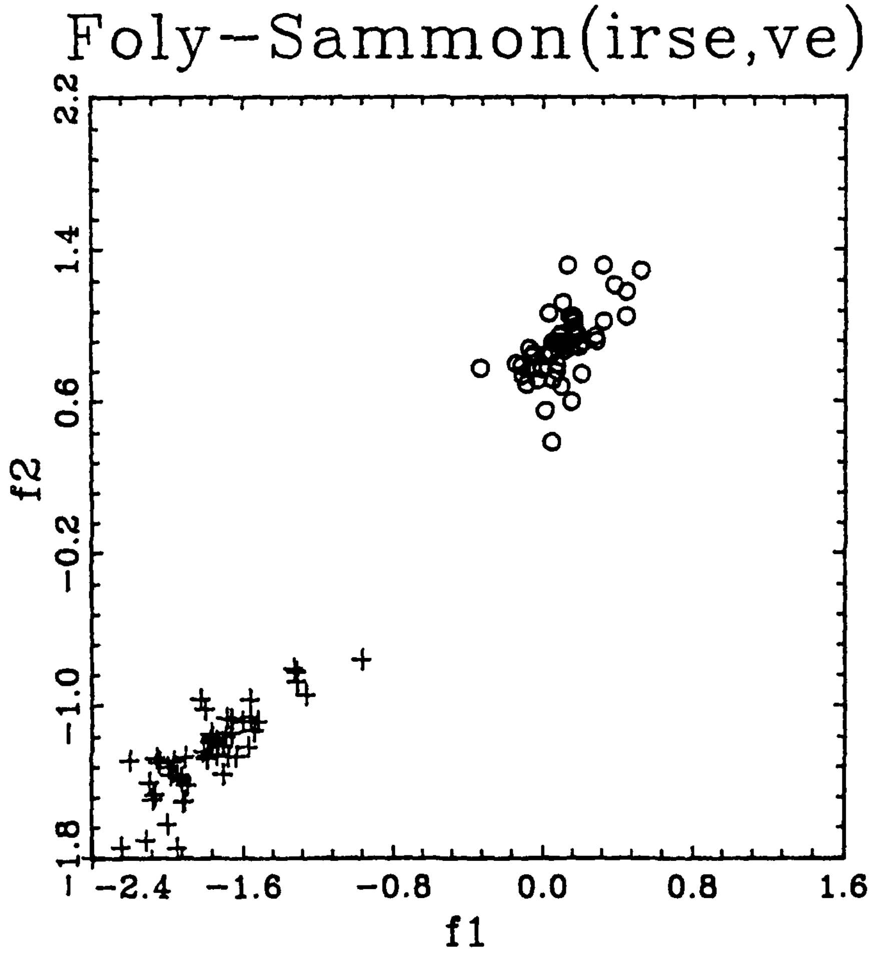


Figure 21 Two clusters of Iris data set obtained by Foley-Sammon algorithm

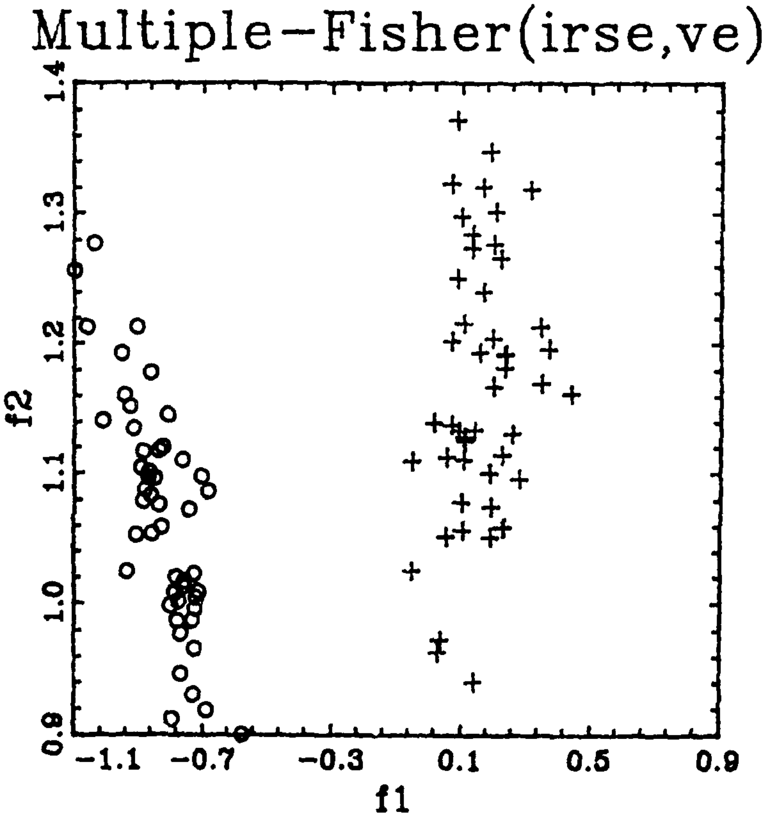


Figure 22 Two clusters of Iris data set obtained by multiple Fisher discriminant algorithm.

Table 1

List of Iris Data Set

Iris setosa				*Iris versicolor*				*Iris virginica*			
Sepal length	*Sepal width*	*Petal length*	*Petal width*	*Sepal length*	*Sepal width*	*Petal length*	*Petal width*	*Sepal length*	*Sepal width*	*Petal length*	*Petal width*
5.1	3.5	1.4	0.2	7.0	3.2	4.7	1.4	6.3	3.3	6.0	2.5
4.9	3.0	1.4	0.2	6.4	3.2	4.5	1.5	5.8	2.7	5.1	1.9
4.7	3.2	1.3	0.2	6.9	3.1	4.9	1.5	7.1	3.0	5.9	2.1
4.6	3.1	1.5	0.2	5.5	2.3	4.0	1.3	6.3	2.9	5.6	1.8
5.0	3.6	1.4	0.2	6.5	2.8	4.6	1.5	6.5	3.0	5.8	2.2
5.4	3.9	1.7	0.4	5.7	2.8	4.5	1.3	7.6	3.0	6.6	2.1
4.6	3.4	1.4	0.3	6.3	3.3	4.7	1.6	4.9	2.5	4.5	1.7
5.0	3.4	1.5	0.2	4.9	2.4	3.3	1.0	7.3	2.9	6.3	1.8
4.4	2.9	1.4	0.2	6.6	2.9	4.6	1.3	6.7	2.5	5.8	1.8
4.9	3.1	1.5	0.1	5.2	2.7	3.9	1.4	7.2	3.6	6.1	2.5
5.4	3.7	1.5	0.2	5.0	2.0	3.5	1.0	6.5	3.2	5.1	2.0
4.8	3.4	1.6	0.2	5.9	3.0	4.2	1.5	6.4	2.7	5.3	1.9
4.8	3.0	1.4	0.1	6.0	2.2	4.0	1.0	6.8	3.0	5.5	2.1
4.3	3.0	1.1	0.1	6.1	2.9	4.7	1.4	5.7	2.5	5.0	2.0
5.8	4.0	1.2	0.2	5.6	2.9	3.6	1.3	5.8	2.8	5.1	2.4
5.7	4.4	1.5	0.4	6.7	3.1	4.4	1.4	6.4	3.2	5.3	2.3
5.4	3.9	1.3	0.4	5.6	3.0	4.5	1.5	6.5	3.0	5.5	1.8
5.1	3.5	1.4	0.3	5.8	2.7	4.1	1.0	7.7	3.8	6.7	2.2
5.7	3.8	1.7	0.3	6.2	2.2	4.5	1.5	7.7	2.6	6.9	2.3
5.1	3.8	1.5	0.3	5.6	2.5	3.9	1.1	6.0	2.2	5.0	1.5
5.4	3.4	1.7	0.2	5.9	3.2	4.8	1.8	6.9	3.2	5.7	2.3
5.1	3.7	1.5	0.4	6.1	2.8	4.0	1.3	5.6	2.8	4.9	2.0
4.6	3.6	1.0	0.2	6.3	2.5	4.9	1.5	7.7	2.8	6.7	2.0
5.1	3.3	1.7	0.5	6.1	2.8	4.7	1.2	6.3	2.7	4.9	1.8
4.8	3.4	1.9	0.2	6.4	2.9	4.3	1.3	6.7	3.3	5.7	2.1
5.0	3.0	1.6	0.2	6.6	3.0	4.4	1.4	7.2	3.2	6.0	1.8
5.0	3.4	1.6	0.4	6.8	2.8	4.8	1.4	6.2	2.8	4.8	1.8
5.2	3.5	1.5	0.2	6.7	3.0	5.0	1.7	6.1	3.0	4.9	1.8
5.2	3.4	1.4	0.2	6.0	2.9	4.5	1.5	6.4	2.8	5.6	2.1
4.7	3.2	1.6	0.2	5.7	2.6	3.5	1.0	7.2	3.0	5.8	1.6
4.8	3.1	1.6	0.2	5.5	2.4	3.8	1.1	7.4	2.8	6.1	1.9
5.4	3.4	1.5	0.4	5.5	2.4	3.7	1.0	7.9	3.8	6.4	2.0
5.2	4.1	1.5	0.1	5.8	2.7	3.9	1.2	6.4	2.8	5.6	2.2
5.5	4.2	1.4	0.2	6.0	2.7	5.1	1.6	6.3	2.8	5.1	1.5
4.9	3.1	1.5	0.2	5.4	3.0	4.5	1.5	6.1	2.6	5.6	1.4
5.0	3.2	1.2	0.2	6.0	3.4	4.5	1.6	7.7	3.0	6.1	2.3
5.5	3.5	1.3	0.2	6.7	3.1	4.7	1.5	6.3	3.4	5.6	2.4
4.9	3.6	1.4	0.1	6.3	2.3	4.4	1.3	6.4	3.1	5.5	1.8
4.4	3.0	1.3	0.2	5.6	3.0	4.1	1.3	6.0	3.0	4.8	1.8
5.1	3.4	1.5	0.2	5.5	2.5	4.0	1.3	6.9	3.1	5.4	2.1
5.0	3.5	1.3	0.3	5.5	2.6	4.4	1.2	6.7	3.1	5.6	2.4
4.5	2.3	1.3	0.3	6.1	3.0	4.6	1.4	6.9	3.1	5.1	2.3
4.4	3.2	1.3	0.2	5.8	2.6	4.0	1.2	5.8	2.7	5.1	1.9
5.0	3.5	1.6	0.6	5.0	2.3	3.3	1.0	6.8	3.2	5.9	2.3
5.1	3.8	1.9	0.4	5.6	2.7	4.2	1.3	6.7	3.3	5.7	2.5
4.8	3.0	1.4	0.3	5.7	3.0	4.2	1.2	6.7	3.0	5.2	2.3
5.1	3.8	1.6	0.2	5.7	2.9	4.2	1.3	6.3	2.5	5.0	1.9
4.6	3.2	1.4	0.2	6.2	2.9	4.3	1.3	6.5	3.0	5.2	2.0
5.3	3.7	1.5	0.2	5.1	2.5	3.0	1.1	6.2	3.4	5.4	2.3
5.0	3.3	1.4	0.2	5.7	2.8	4.1	1.3	5.9	3.0	5.1	1.8

6. Concluding Remarks and Future Developments

The IUNDE package is developed with the primary intentions of being very easy to use and that the user can familiarize himself with all these operations very quickly. It is our belief that this package can serve as a very effective tool that assists engineers and researchers in practical applications to NDE problems.

The IBM-PC (or compatibles) version of this package has a very powerful graphics capability. Many different kinds of peripherals are allowed. The user can run the **EQUIP.exe** program to see the details.

A second version of the IUNDE package is being developed by Information Research Lab., Inc. and is aimed towards the following goals :

1. Expansion of the signal processing library by including such functions as Spectral Zooming and Spectral Expansion, Bicorrelational Bispectral Analysis and Model-Based Bispectral Analysis, Spectral Extrapolation and L2 Deconvolutions, etc.

2. Integrate with the RCE™ Neual Network developed by Nestor® Inc. to perform more reliable and versatile pattern classifications.

3. Include the ability to "talk" with some specific ultrasonic hardwares such as pulser/receiver and high speed digitizer to build up an on-line defect classification system.

4. Expansion of the graphic display fuctions by adding the options for 3-dimensional surface plots, contour plots and multi-window graphic displays.

5. Enhancement of the user interface; the new version IUNDE is all menu-driven with pop-up dialog box which adds to the use of this package a lot more fun.

Appendix A

References

[1] C.H. Chen and W.L. Hsu, " *Modern spectral analysis for Ultrasonic Nondestructive Testing* ",Conference on NDT ofHigh-Performance Ceramics, August 25-27, 1987, Boston, MA.

[2]C.H. Chen, *Nonlinear Maximum Entropy Spectral Analysis Methods for Signal Recognition*, Research Studies Press, Wiley, 1982.

[3]J.W. Sammon, "*A nonlinear mapping for data structure analysis* ", IEEE Transactions on Computer, May 1969.

[4]Sophocles J. Orfanidis, *Optimum Signal Processing*, McMillan Publishing Co., 1984.

[5]C.H. Chen, W.L. Hsu and S.K. Sin, " *A comparison of wavelet deconvoltuion techniques for ultrasonic nondestructive testings* ", Proceedings of the 1988 IEEE International Conference on ASSP, pp.867-870, April 1988.

[6]R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, Wiley 1972.

[7]R.C. Gonzalez and J.T. Tou, *Pattern Recognition Principles*, Addison-Wesley Publishing Co., 1974.

[8]D.H. Foley and J.W. Sammon, " *An optimum set of discriminant vectors* ", IEEE Transactions on Computer, March 1975.

[9]C.H. Chen, " *HIgh resolution spectral analysis NDE techniques for flaw characterization, prediction and discrimination* " in **Signal Processing and Pattern Recognition in NDE of**

Materials by C.H. Chen (Ed.), Springer Verlag, NATO ASI Series, 1988.

[10]C.H. Chen," *A signal processing study of ultrasonic nondestructive evaluation of materials* ", Technical Report No. MTL TR 87-11, Material Technology Lab., Watertown, MA., Feb. 1987.

[11] P. Flandrin," *Non-destructive evaluation in the time-frequency domain by means of WIgner-Ville Distribution* ", in **Signal Processing and Pattern Recognition in NDE of Materials** by C.H. Chen (Ed.), Springer Verlag, NATO ASI Series, 1988.

Appendix B : Ultrasonic Waveforms

We have included some ultrasonic pulse echo samples in this package. All these signals are sampled with a sampling rate of 100 MHz. The names of these signals should be interpreted as follows :

(1)T5 or T15 indicates the type of ultrasonic transducers used for acquiring these signals. T5 indicates a 5 MHz transducer and T15 indicates a 15 MHz transducer.

(2)A, B, C and D indicate the relative sizes of the hidden defects in the aluminium blocks from which these pulse echoes are obtained; such that

sizes of A : B : C: D = 8 : 4 : 2 : 1

(3)A0 : no hidden defects (a flawless sample).

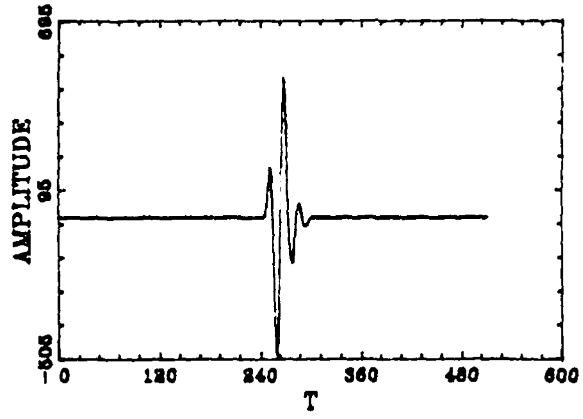
A1, B1, C1 and D1 : hidden defects are flat-topped cuts.

A2, B2, B3, C2, C3,
D2, D3 and D4 : hidden defects are angular cuts.

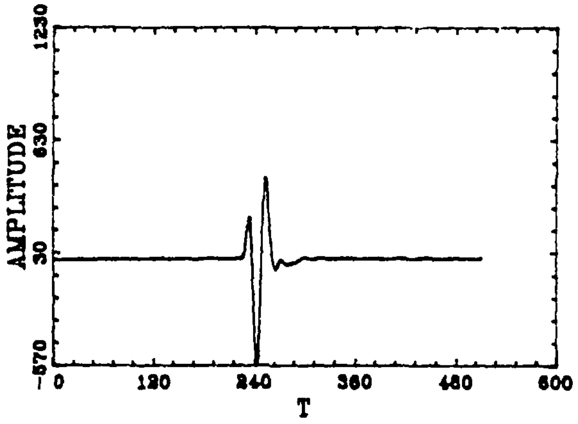
A3, B4, C4 and D5 : hidden defects are circular holes.

Appendix B: Ultrasonic Signal Samples

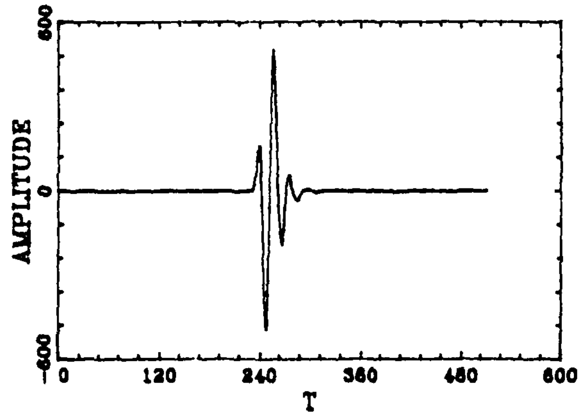
T5A0



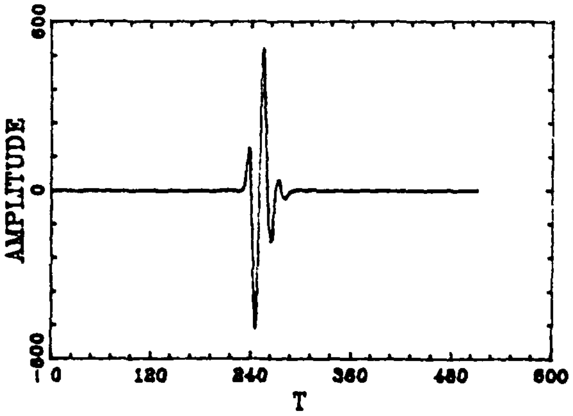
T5A3



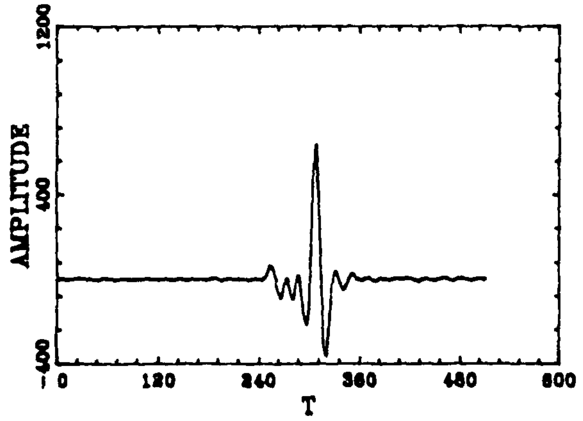
T5A1



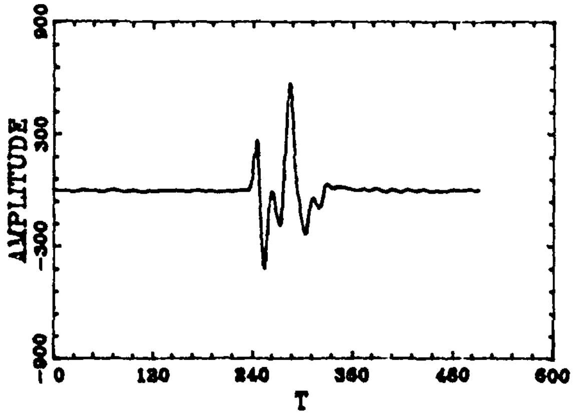
T5B1



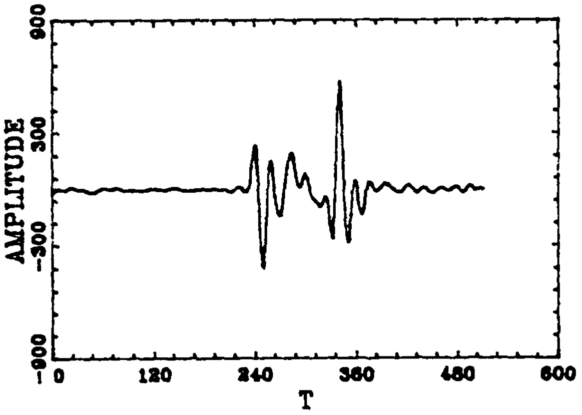
T5A2



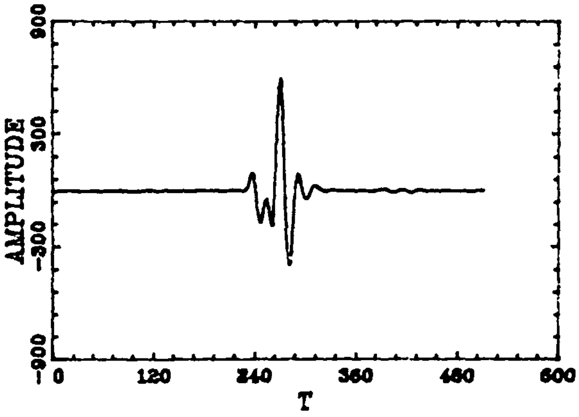
T5B2



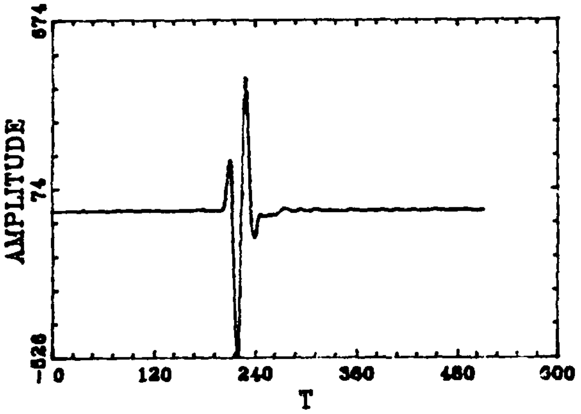
T5B3



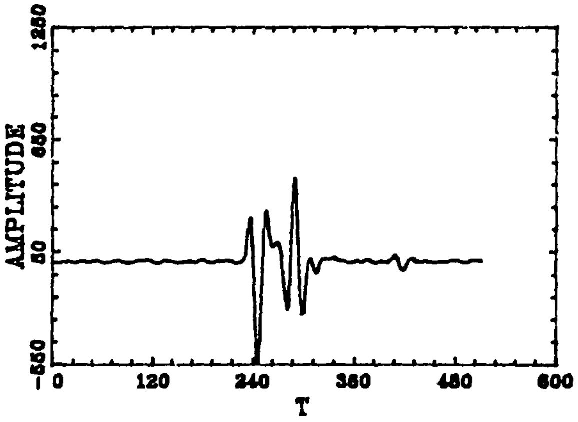
T5C2



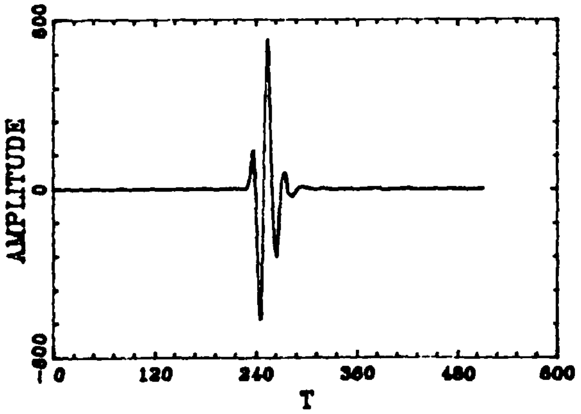
T5B4



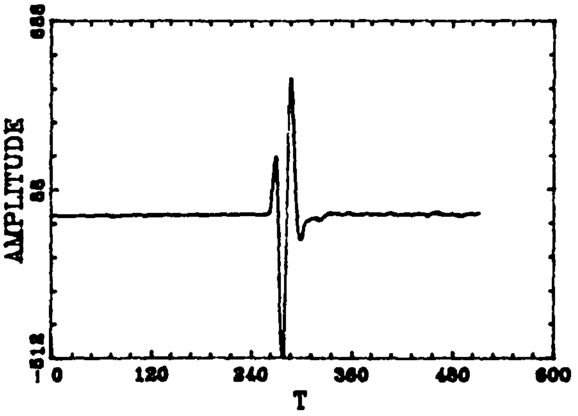
T5C3



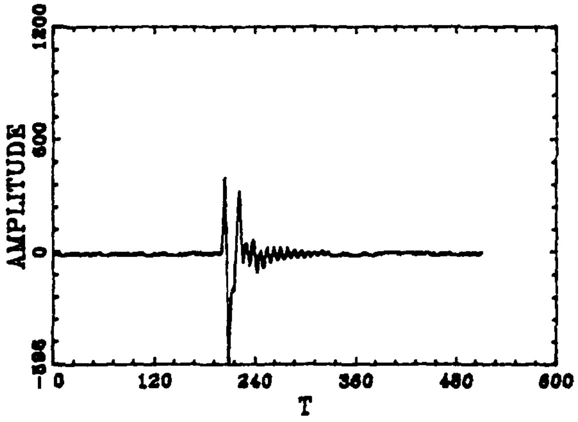
T5C1



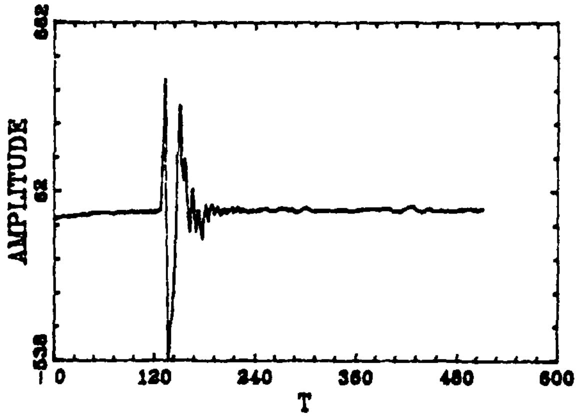
T5C4



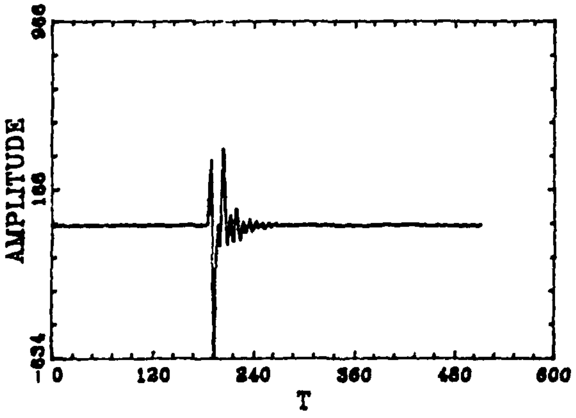
T15A0



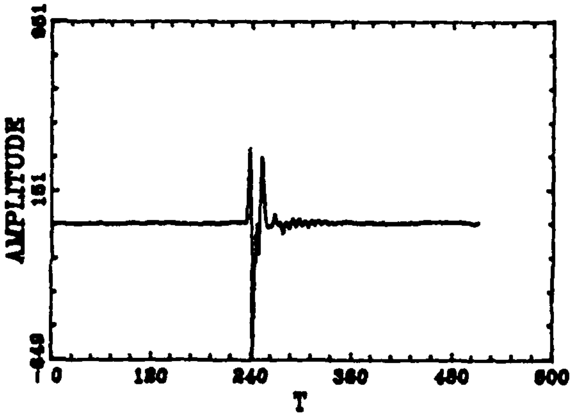
T15A3



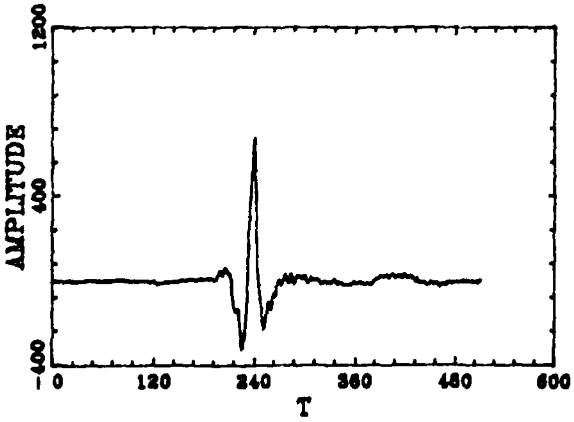
T15A1



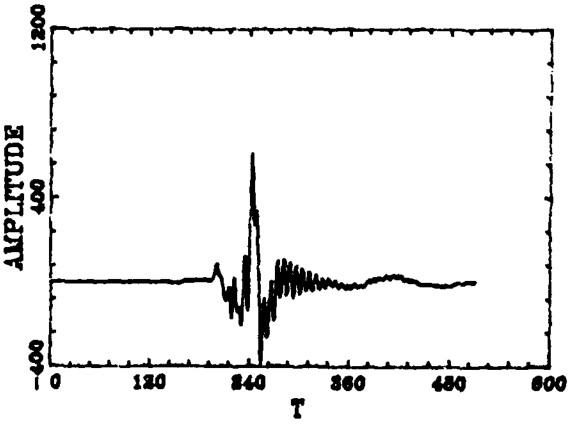
T15B1



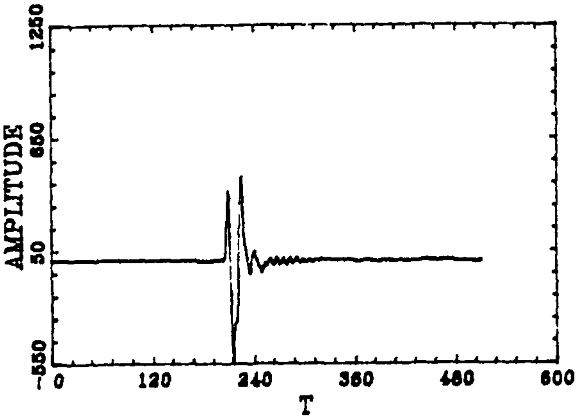
T15A2



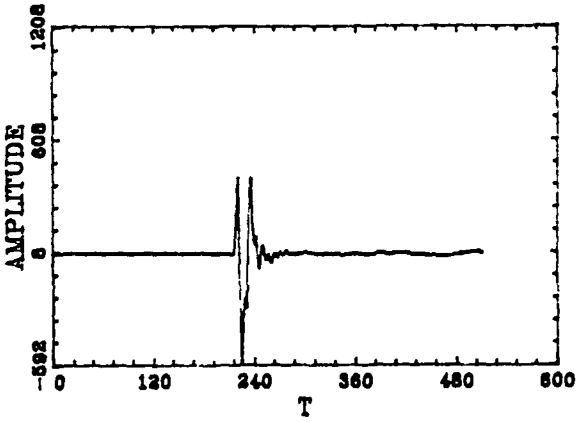
T15B2



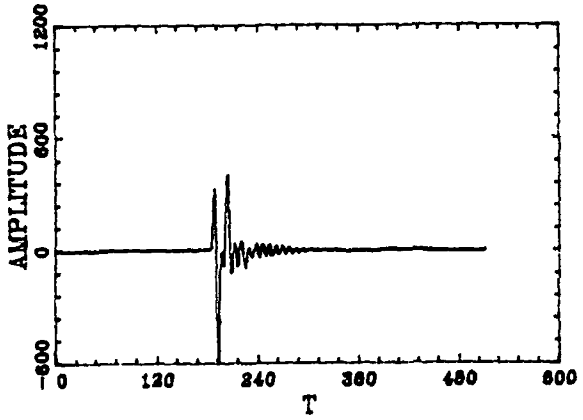
T15B4



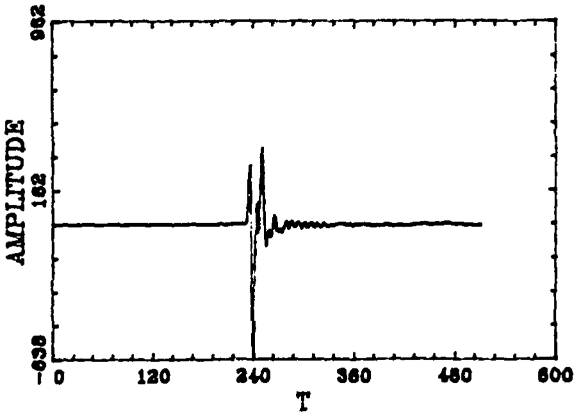
T15C4



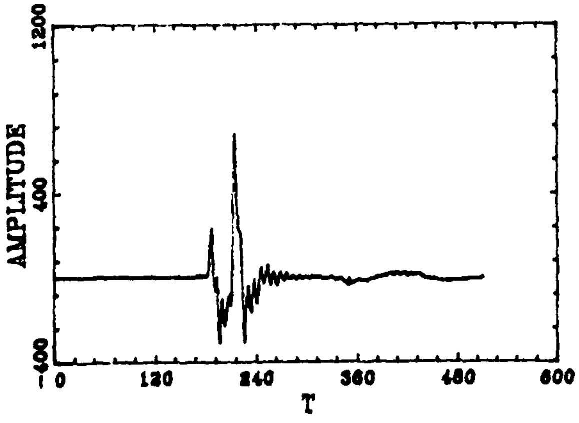
T15C1



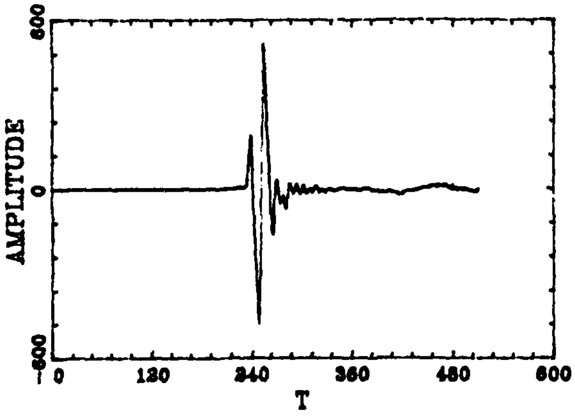
T15D1



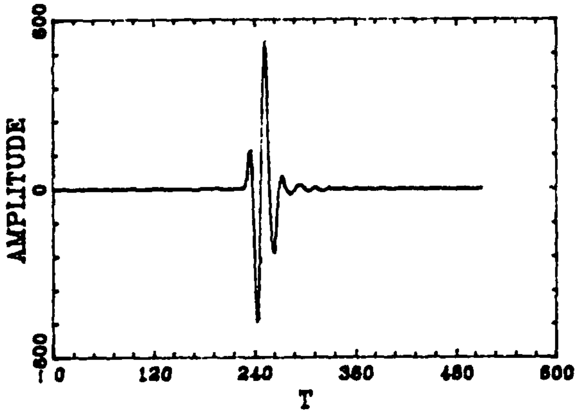
T15C2



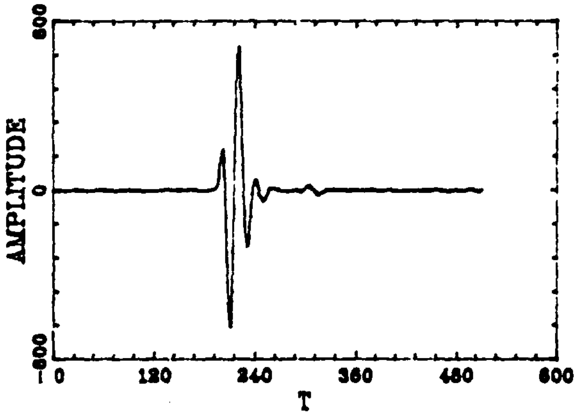
T15D2



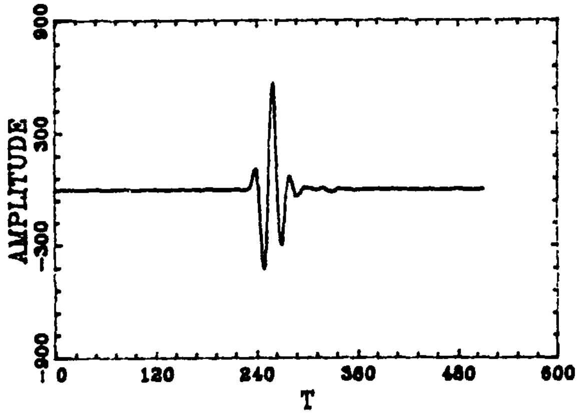
T5D1



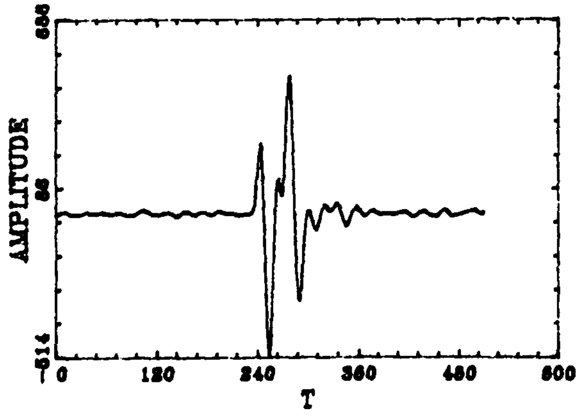
T5D5



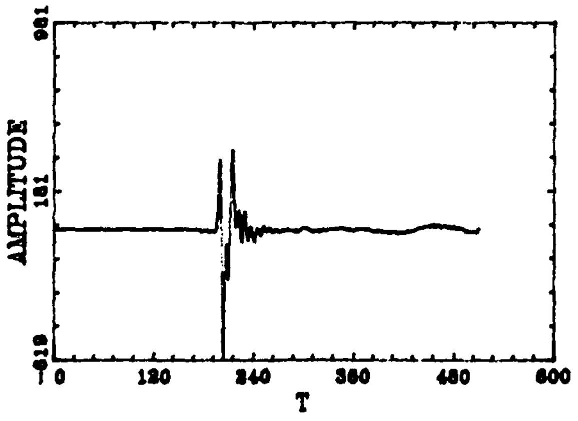
T5D2



T5D3



T15D5



UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER MTL TR 88-34	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) INTERACTIVE ULTRASONIC NONDESTRUCTIVE EVALUATION		5. TYPE OF REPORT & PERIOD COVERED Final - 9/1/87 through 9/30/88
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) C. H. Chen and S. K. Sin		8. CONTRACT OR GRANT NUMBER(s) DAAL04-88-C-0003
9. PERFORMING ORGANIZATION NAME AND ADDRESS Information Research Laboratory, Inc. 415 Bradford Place North Dartmouth, MA 02747		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Materials Technology Laboratory ATTN: SLCMT-PR Watertown, MA 02172-0001		12. REPORT DATE October 1988
		13. NUMBER OF PAGES 48
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

COR: Otto R. Gericke

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Ultrasonic signals
Nondestructive evaluation
Wiener filtering
Material testing
Personal computer
Burg's power spectrum
Cross-correlation
Feature extraction
Software package

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

(SEE REVERSE SIDE)

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block No. 20

ABSTRACT

Modern DSP (Digital Signal Processing) and Pattern Recognition techniques are very effective tools for Ultrasonic NDE (Non-Destructive Evaluations) of materials. In recent researches, such techniques as High Resolution Spectral Analysis, Deconvolution, K-mean Clustering, Non-linear Mapping and Multiple Fisher's Discriminant, have been applied to the classification of hidden defect geometries with high accuracy.

DISTRIBUTION LIST

No. of Copies — To

Metals and Ceramics Information Center, Battelle Columbus Laboratories, 505 King Avenue, Columbus, OH 43201
1 ATTN: Mr. Harold Mindlin, Director
1 Mr. James Lynch, Assistant Director

DTIC-FDAC Commander, Defense Technical Information, Center, Cameron Station, Building 5, 5010 Duke Street, Alexandria, VA 22314
2 ATTN: DTIC-FDAC

Commander, U.S. Army Foreign Science and Technology Center, 220 Seventh St., N.E., Charlottesville, VA 22901
1 ATTN: AMXST-SD3

Commander, Army Research Office, P.O. Box 12211, Research Triangle Park, NC 27709-2211
1 ATTN: Mr. J. J. Murray

Commander, U.S. Army Materiel Command, 5001 Eisenhower Avenue, Alexandria, VA 22333
1 ATTN: AMCQA-E
1 AMCQA-P
1 AMCDE-D
1 AMCMT
1 AMCMM-M

Commander, U.S. Army Laboratory Command, 2800 Powder Mill Road, Adelphi, MD 20783-1197
1 ATTN: AMSLC-IM-TL

Commander, U.S. Army Electronics Research and Development Command, Fort Monmouth, NJ 07703
1 ATTN: AMDSD-PA-E, Mr. Stan Alster
1 AMDSD-PA-E, Mr. J. Quinn

Commander, U.S. Army Missile Command, Redstone Arsenal, AL 35898-5241
2 ATTN: AMSMI-TB Redstone Scientific Information Center
1 AMSMI-TK, Mr. J. Alley
1 AMSMI-M
1 AMSMI-ET, Mr. Robert O. Black
1 AMSMI-QS, Mr. George L. Stewart, Jr.
1 AMSMI-EAT, Mr. R. Tailey
1 AMSMI-QP
1 AMSMI-RLM

Commander, U.S. Army Materiel Systems Analysis Activity, Aberdeen Proving Ground, MD 21005
1 ATTN: AMXSY-MP, H. Cohen

Director, U.S. Army Ballistic Research Laboratory, Aberdeen Proving Ground, MD 21005
1 ATTN: SLCBR-TSB-S (STINFO)

Commander, U.S. Army Troop Support and Aviation Materiel Readiness Command, 4300 Goodfellow Boulevard, St. Louis, MO 63120
1 ATTN: AMSTS-PLE(2), Mr. J. Corwin
1 AMSTS-Q
1 AMSTS-M

Commander, U.S. Army Natick Research, Development and Engineering Center, Natick, MA 01760
1 ATTN: AMDNA-EM

Commander, Chemical Systems Laboratory, Aberdeen Proving Ground, MD 21010
1 ATTN: AMSME-CLD, Mr. W. E. Montanary

Commander, U.S. Army Mobility Equipment Research and Development Command, Fort Belvoir, VA 22060
1 ATTN: AMDME-D
1 AMDME-E
1 AMDME-G
1 AMDME-H
1 AMDME-M
1 AMDME-T
1 AMDME-V
1 AMDME-ZE
1 AMDME-N

Commander, U.S. Army Armament, Munitions and Chemical Command, Rock Island, IL 61299-6000
2 ATTN: AMSAR-QA
1 AMSAR-SC
1 AMSAR-RDP
1 AMSAR-EN
1 AMSAR-QAE

Commander, Rock Island Arsenal, Rock Island, IL 61299
1 ATTN: SARRI-EN, Mr. W. M. Kisner
SARRI-ENM, W.D. McHenry
SARRI-QA

Commander, U.S. Army Armament, Munitions and Chemical Command, Dover, NJ 07801
1 ATTN: AMDAR-LC, Mr. E. Kelly
1 AMDAR-LCE, Dr. Walker
1 AMDAR-TSP, Mr. B. Stephans
2 AMDAR-TSS (STINFO)
1 AMDAR-LCA, Mr. Harry E. Pebly, Jr., PLASTEC, Director

Commander, AMCCOM, Product Assurance Directorate, Aberdeen Proving Ground MD 21010
1 ATTN: AMDAR-QAC-E, Dr. W. J. Maurits

Commander, Watervliet Arsenal, Watervliet, NY 12189
1 ATTN: AMSMC-LCB, Mr. T. Moraczewski
2 SARWV-PPI, Mr. L. Jette

Commander, U.S. Army Aviation Systems Command, AVSCOM, St. Louis, MO 63120
1 ATTN: AMSAV-EGX
AMSAV-QR
AMSAV-QP
AMSAV-QE

Commander, U.S. Army Tank-Automotive Command, Warren, MI 48090
1 ATTN: AMSTA-UL, Technical Library
1 AMSTA-RCKM, Mr. S. Goodman
1 AMSTA-RCKT, Mr. J. Fix
1 AMSTA-RTAS, Mr. S. Catalano
1 AMSTA-TTM, Mr. W. Moncrief
1 AMSTA-ZS, Mr. O. Renius
1 AMSTA-JA, Mr. C. Kedzior
2 AMSTA-Q

Director, U.S. Army Industrial Base Engineering Activity, Rock Island, IL 61299
1 ATTN: AMXIB-MT, Mr. D. Brim

Commander, Harry Diamond Laboratories, 2800 Powder Mill Road, Adelphi, MD 20
1 ATTN: SLCHD-EDE, Mr. B. F. Willis

Commander, U.S. Army Test and Evaluation Command, Aberdeen Proving Ground, MD 21005
1 ATTN: AMSTE-TD
1 AMSTE-ME

Commander, U.S. Army White Sands Missile Range, NM 88002
1 ATTN: STEWS-AD-L
1 STEWS-ID
1 STEWS-TD-PM

Commander, U.S. Army Yuma Proving Ground, Yuma, AZ 85364
1 ATTN: Technical Library

Commander, U.S. Army Tropic Test Center, Fort Clayton, Canal Zone
1 ATTN: STETC-TD, Drawer 942

Commander, Aberdeen Proving Ground, MD 21005
1 ATTN: STEAP-MT
1 STEAP-MT-M, Mr. J. A. Feroli
1 STEAP-MT-G, Mr. R. L. Huddleston

Commander, U.S. Army Cold Region Test Center, APO Seattle, WA 98733
1 ATTN: STECR-OP-PM

Commander, U.S. Army Dugway Proving Ground, Dugway, UT 84022
1 ATTN: STEDP-MT

Commander, U.S. Army Electronic Proving Ground, Fort Huachuca, AZ 85613
1 ATTN: STEEP-MT

Commander, Jefferson Proving Ground, Madison, IN 47250
1 ATTN: STEJP-TD-1

Commander, U.S. Army Aircraft Development Test Activity, Fort Rucker, AL 36362
1 ATTN: STEBG-TD

President, U.S. Army Armor and Engineer Board, Fort Knox, KY 40121
1 ATTN: ATZKOAE-TA

President, U.S. Army Field Artillery Board, Fort Sill, OK 73503
1 ATTN: ATZR-BDOP

Commander, Anniston Army Depot, Anniston, AL 36202
1 ATTN: SDSAN-QA

Commander, Corpus Christi Army Depot, Corpus Christi, TX 78419
1 ATTN: SDSCC-MEE, Mr. Haggerty, Mail Stop 55

Commander, Letterkenny Army Depot, Chambersburg, PA 17201
1 ATTN: SDSLE-QA

DISTRIBUTION LIST

No. of Copies — To

Commander, Lexington-Bluegrass Army Depot, Lexington, KY 40507
1 ATTN: SDSLX-QA

Commander, New Cumberland Army Depot, New Cumberland, PA 17070
2 ATTN: SDSNC-QA

Commander, U.S. Army Depot Activity, Pueblo, CO 81001
2 ATTN: SDSTE-PU-Q

Commander, Red River Army Depot, Texarkana, TX 75501
1 ATTN: SDSRR-QA

Commander, Sacramento Army Depot, Sacramento, CA 95813
1 ATTN: SDSSA-QA

Commander, Savanna Army Depot Activity, Savanna, IL 61074
1 ATTN: SDSSV-S

Commander, Seneca Army Depot, Romulus, NY 14541
1 ATTN: SDSSE-R

Commander, Sharpe Army Depot, Lathrop, CA 95330
2 ATTN: SDSSH-QE

Commander, Sierra Army Depot, Herlong, CA 96113
1 ATTN: SDSSI-DQA

Commander, Tobyhanna Army Depot, Tobyhanna, PA 18466
1 ATTN: SDSTO-Q

Commander, Tooele Army Depot, Tooele, UT 84074
1 ATTN: SDSTE-QA

Director, AMC Ammunition Center, Savanna, IL 61074
1 ATTN: SARAC-DE

Naval Research Laboratory, Washington, DC 20375
2 ATTN: Code 5830
2 Library, Code 2620

Commander, U.S. Air Force Wright Aeronautical Laboratories,
Wright Patterson Air Force Base, OH 45433
2 ATTN: AFWAL/MLTM, Mr. W. Wheeler
2 AFWAL/MLLP, Mr. R. Rowand

1 Mr. R.J. Zentner, EAI Corporation, 626 Towne Center Drive, Suite 205, Joppatowne,
MD 21085-4440

Director, U.S. Army Materials Technology Laboratory, Watertown, MA 02172-0001
2 ATTN: SLCMT-TML
1SLCMT-IMA-T
1SLCMT-PR
1SLCMT-MRM, Gericke