AD-A259 774

SRINCON CORPORATION

OCR 93-U-0029

PHASE I CONDITION BASED MACHINERY MAINTENANCE (HELICOPTER FAULT DETECTION)

Final Report

29 January 1993

DTIC
ELECTE
FEB 0 2 1993
S B D

Contract No. N00014-92-C-0059
CDRL No. 0001AC

Prepared by:
T. W. Brotherton and T. G. Pollard

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

Submitted to:
Scientific Officer
Office of Naval Research
800 North Quincy Street
Arlington, VA 22217-5000
Attn: Dr. Thomas McKenna

93-01832

Distribution:

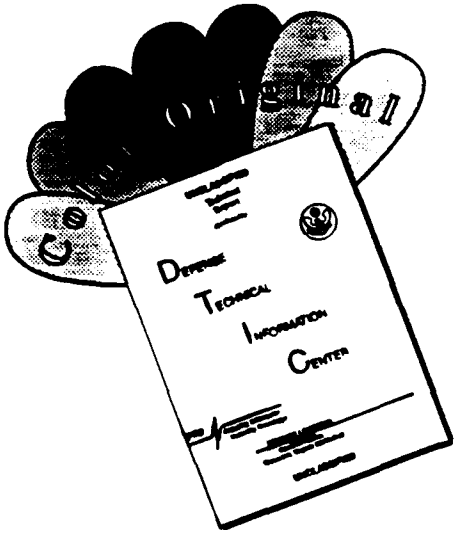
Scientific Officer	N00014	1/0
Admin. Contracting Officer	S0514A	1/0
Director, NRL	N00173	1/0
DTIC	S47031	2/0

Original contains color plates: All DTIC reproductions will be in black and white.

93 2 1 046

9363 Towne Centre Drive, San Diego, CA 92121 Phone: (619) 455-5530 FAX #: (619) 453-9274

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF COLOR PAGES WHICH DO NOT REPRODUCE LEGIBLY ON BLACK AND WHITE MICROFICHE.

Table of Contents

1.0 INTRODUCTION 1

2.0 APPROACH 3
- 2.1 Hierarchical Neural Net Training and Testing 5
- 2.2 Feature Extraction 5
- 2.3 Neural Network Classifier 9

3.0 REAL DATA PROCESSING RESULTS 9
- 3.1 Hollins Helicopter Gearbox Data 9
- 3.2 Fire Pump Data 23
- 3.3 Condenser Pump Data 27

4.0 CONCLUSIONS 28

5.0 RECOMMENDATIONS 31

APPENDIX A A-1

List of Tables

Table 1 Single-Channel Processing Parameters 14
Table 2 "Multichannel" Processing Parameters 17
Table 3 Hollins Data Confusion Matrix Trained Without Cut 8 22
Table 4 Hollins Data Performance Summary 22
Table 5 Fire Pump Multichannel Processing Parameters 23
Table 6 Fire Pump System Performance Summary 27
Table 7 Condenser Pump Processing Parameters 27
Table 8 Condenser Pump System Performance Summary 31

List of Figures

Figure 1 Multifeature Network Proposed for Phase I 4
Figure 2 Hierarchical Neural Network Architecture Used in Phase I 4
Figure 3 Hierarchical Neural Network Training and Test Flow Diagram 6
Figure 4 Three-Layered Perceptron Neural Network Used by ORINCON for Feature Space Detection and Classification 10
Figure 5 Single-Channel, Multifeature Processing of Hollins Data Set 10
Figure 6 Bearing, Outer Race Fault in Hollins Multifeature System 12
Figure 7 Inputs and Outputs for the First-Layer Nets 13
Figure 8 AOK TFR Applied to Hollins Cut-Tooth Data 15
Figure 9 Inputs and Outputs for the Second-Layer Nets 17
Figure 10 "Multichannel" Processing Flow Diagram for the Hollin's Data Set Full 18
Figure 11 Bearing Inner Race Fault Example 19
Figure 12 Bearing Rolling Element Fault Condition 21
Figure 13 Fire Pump and Condenser Pump Multichannel Processing Flow Diagram 23
Figure 14 Fire Pump System with Class 1 Fault 25
Figure 15 Fire Pump System with Normal Data 26
Figure 16 Condenser Pump System with Class 2 Fault 29
Figure 17 Condenser Pump System with Normal Data 30
Figure 18 Phase II System for Helicopter Transmission Fault Detection and Classification 33

For
I ☑
☐
d ☐
ion
per
ADA255796
n/
ty Codes
and/or
cial
DTIC QUALITY INSPECTED 3
A-1

ORINCON CORPORATION

1.0 INTRODUCTION

A problem of current interest to the Navy is the automatic detection and classification of faults in mechanical systems such as the transmissions, gearboxes, and bearings of helicopters. The problem is important for both economic and safety reasons. Using automated fault detection and classification, machinery repair can be undertaken as needed and can prevent catastrophic failure. This *condition-based* maintenance approach is more efficient and cost-effective than the use of predetermined maintenance schedules. Additionally, automated fault detection and classification systems can alert machinery operators, such as helicopter pilots, of the onset of a mechanical problem, thus allowing the pilot to take precautions, such as landing the aircraft, before a potentially catastrophic failure occurs.

The machinery fault detection and classification problem is general in nature, and the methodology developed in Phase I is applicable to a variety fault detection scenarios involving multichannel and multisensor time series sensor data. Current helicopter gearbox fault detection techniques make use of metal particle detectors in the transmission oil as well as vibration sensors. For the vibration data, relatively simple metrics and thresholds are specified using analytic models for the gearboxes and bearings. Unfortunately, the performance of this approach is only as good as the model developed. In many cases, the interaction of fault conditions with the mechanical system is time varying and highly nonlinear so that anticipated signatures may not be seen[1]. Specifying an accurate model is at best difficult and often impossible. Additionally, such model-based processing is inherently inflexible, being applicable only to the specific system of interest. Model-based fault detection and classification algorithms cannot easily be adapted to accommodate machinery (transmission) design changes, new fault detection tasks, or the addition of new sensor types.

For the Phase I effort, we have developed a variety of neural networks, coupled with spectral feature extractors, to solve the fault detection and classification problem. A novel, hierarchical neural network architecture was developed that fuses multifeature and multichannel accelerometer vibrational sensor data to successfully detect and classify fault conditions. The approach was successfully applied to three different types of rotating machinery (helicopter gearboxes, fire pumps, and condenser pumps). The approach also allows for the evaluation of the utility of the various feature extractors and for the channels to include in solving the fault detection/classification problem at hand.

The goal of the Phase I research effort was to develop and demonstrate a system that uses multiple feature extractors and a neural net classifier to perform fault detection and classification

[1] D. Rock, D. Malkoff, and R. Stewart, "AI and Aircraft Health Monitoring," AI Expert, Feb. 1993.

for rotating machinery. The systems developed are sensitive to fault conditions and robust with respect to normal operation and fault variations. The primary tasks accomplished in the development and demonstration of the system for Phase I were the following:

- **Identify and develop feature extraction processing.** Several features were examined in Phase I: the short-time Fourier transform (STFT), instantaneous time-frequency representations (TFRs), Prony's model method, and the wavelet transform using the Gabor wavelet filter.
- **Assess the performance of the various feature extractors.** The real accelerometer data was processed using each of the candidate feature extractors, and an evaluation of the utility of each was made.
- **Train the neural network.** Training templates for each fault class were selected using the feature vectors, and multiple three-layer perceptron neural networks were trained with these templates. The outputs from the first-layer nets were merged and used to train a second-layer net using the hierarchical neural net approach[2,3].
- **Test the system.** System performance was assessed by running test accelerometer data through the system and determining the system's probability of fault detection (P_D), probability of false alarm (P_{FA}), probability of correct classification (P_C), and probability of misclassification (P_{MC}).

The approach developed in the Phase I effort was applied to three different data sets. The data sets included two channels of data from the Hollin's data set[4], six channels of data from an aircraft carrier fire pump, and six channels of data from a condenser pump. All the measurements were of mechanical vibrations of the system. The approach developed was easily adapted to handle all of the different types of mechanical systems and vibration measurements. In all cases, the prototype systems developed gave perfect performance for the data sets supplied for the Phase I effort. That is, each system developed gave the correct classifications with no resulting false alarms for all the data sets processed.

It was also found that the simple FFT feature extractor was sufficient to solve the problem for all three of the mechanical systems. The FFT is the most computationally efficient of all the feature extractors considered. This result is important when considering real-time implementation for future development. It was found that the high-resolution time/frequency feature extractors gave relatively poor performance. It is believed that the high-resolution feature extractors give too much detail when included in the overall system. The high-resolution features are not robust features for solving the detection/classification problem considered here; there is too much variation within the data. FFT processing, on the other hand, tends to smear the features so that

---

[2] T. Brotherton, D. Fogel, and E. Mears, "Final Report for Applications of Data Fusion to Signal Processing Phase I SBIR," DARPA order no. 5916, April 1992.

[3] T. Brotherton and E. Mears, "Application of Neural Nets to Feature Fusion," *26th Asilomar Conference on Signals, Systems, and Computers*, October 1992.

[4] M. Hollins, "The Effects of Vibration Sensor Location in Detecting Gear and Bearing Defects," *41st Mechanical Failures Prevention Group Proceedings*, 1988.

much of the detail and variation are lost. The result is a more robust feature for detection and classification.

It was found that multichannel processing gives more robust performance than single-channel processing. By processing multichannel data, the probability of correct classification was increased while the probability of false alarm was simultaneously reduced. In the six-channel pump data, faults in the test data sets did not appear on all of the channels that were used in the training data sets. The channels that the fault did appear on were fault dependent. For the two-channel Hollin's data set, it was found that only a single channel was required to solve the detection/classification problem; the second channel did not add any information to aid detection and classification.

The remainder of this report gives more detailed descriptions and details found in the Phase I effort. Section 2 describes the multifeature/hierarchical neural network approach used for processing of the three data sets. Section 3 gives detailed results for processing of the three data sets. Section 4 gives conclusions and Section 5 gives recommendations. Appendix A is a copy of a conference paper developed under Phase I.

2.0 APPROACH

The original Phase I proposal proposed multifeature fusion using an architecture like that shown in Figure 1. In this system, feature vectors from multiple signal processing techniques are appended into a larger "fusion" vector, and these expanded vectors are used as inputs to a three-layer perceptron neural network for fault detection and classification.

However, subsequent research under DARPA SBIR funding demonstrated the superiority of a hierarchical neural network data fusion architecture. The advancements made in the DARPA program were used as a starting point for the fault detection research. The hierarchical architecture developed and tested in the DARPA program is shown in Figure 2.

In the generic system shown in Figure 2, multiple channels and/or feature extractors feed their own individual first-layer neural networks. These neural nets make their own independent detection and classification decisions based on prior training. The outputs from the first-layer networks are vectors of activation levels. The activation level for each normal or fault class indicates each first-layer net's decision as to the operating condition of the machinery. The output activation vectors from the first-layer nets are then merged (appended) into a fusion vector. The time series of fusion vectors from the first-layer nets are then used as inputs to the second-layer fusion neural network. This network makes the final fault detection and classification determination. In effect, the first-layer neural nets perform a nonlinear filtering

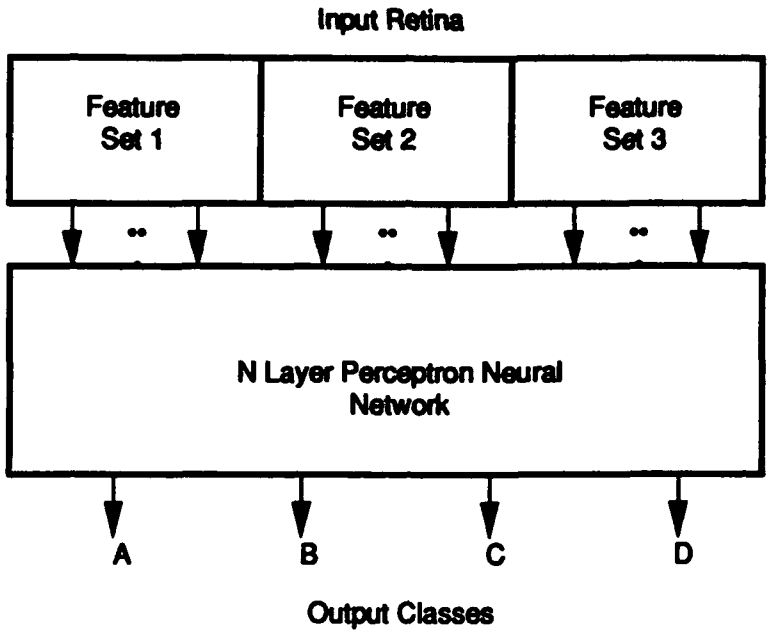


Figure 1. Multifeature Network Proposed for Phase I

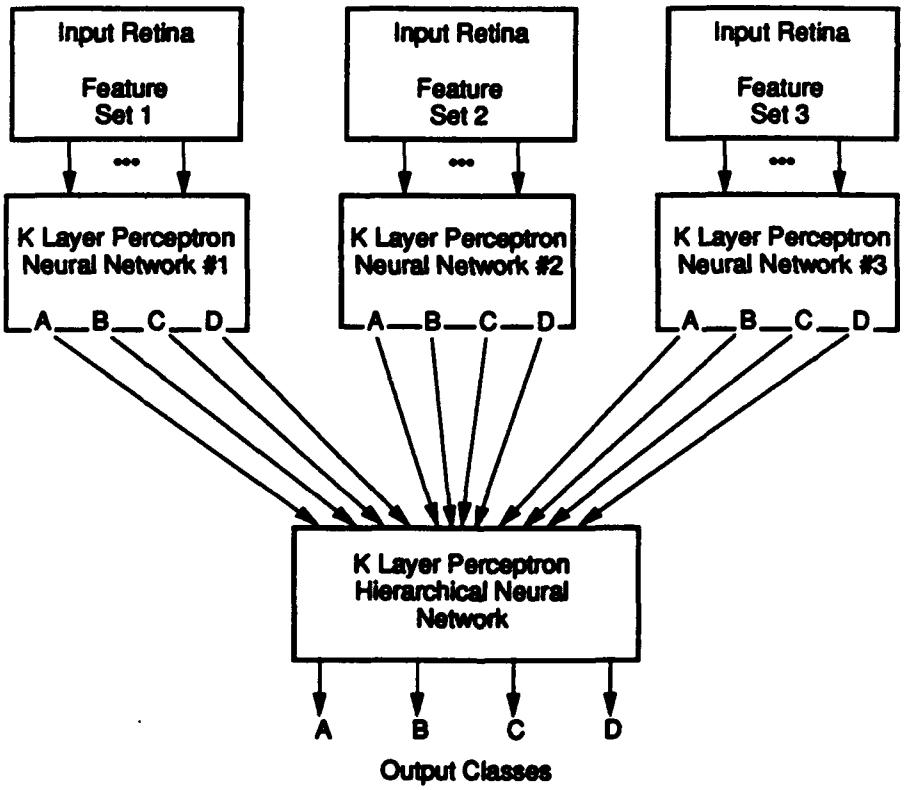


Figure 2. Hierarchical Neural Network Architecture Used in Phase I

operation on the data in order to produce smaller (reduced dimensionality) feature sets. The second-layer net fuses the "filtered" information to solve the detection and classification problem. While the generic architecture of Figure 2 was used for all of the processing in Phase I, each of the individual data sets employed a problem-specific fault detection and classification system. The specific systems used are discussed in Section 3.0.

2.1 Hierarchical Neural Net Training and Testing

Figure 3 shows a high-level training and testing flow diagram for the hierarchical neural network. Initially, the accelerometer data is partitioned to develop training and testing data sets. The training data is then run through each of the different feature extractors, as shown in Figure 2. The output feature space representations are examined and exemplars representative of the events of interest are selected for each feature extractor.

The single-feature exemplars are then used to train the set of first-layer single-feature neural networks. To finish the training of the hierarchical architecture, each of the trained first-layer neural nets is rerun on the training data. The output "activation" level vectors from each network (indicating the input data's degree of membership for each fault class) are used to form a single fusion vector; that is, the various outputs are aggregated together to form the "features" for input to the second-layer neural net. A second exemplar selection is then performed to derive exemplars with which to train the second-layer network. This fusion net is then trained with the fusion exemplar vectors. Finally, the entire architecture is tested by inputting the test data. The resulting fault detection and classification is determined by comparing the class activation levels at the output of the fusion net.

2.2 Feature Extraction

A major issue in the design of a fault detection and classification system is the selection of appropriate features. The goal in selecting features is to choose a set that can characterize signals of interest sufficiently so that classes are well separated when solving the classification problem. ORINCON has investigated a large variety of feature extraction algorithms[5]. The feature extractors investigated and developed for the Phase I effort are discussed below.

- **Short-Time Fast Fourier Transform (STFT):** The short-time fast Fourier transform was a good candidate technique because of its ease of implementation. For real-time applications, fast algorithms and fast chips for hardware realizations are available. Transition from a prototype system to a dedicated hardware system is low cost. The

[5] T. Brotherton, T. Pollard, R. Barton, A. Krieger, L. Marple, "Application of Time-Frequency and Time-Scale Analysis to Underwater Acoustic Transients," *Proc. IEEE International Symposium on Time-Frequency and Time-Scale Analysis*, Victoria, B.C., Canada, October, 1992.

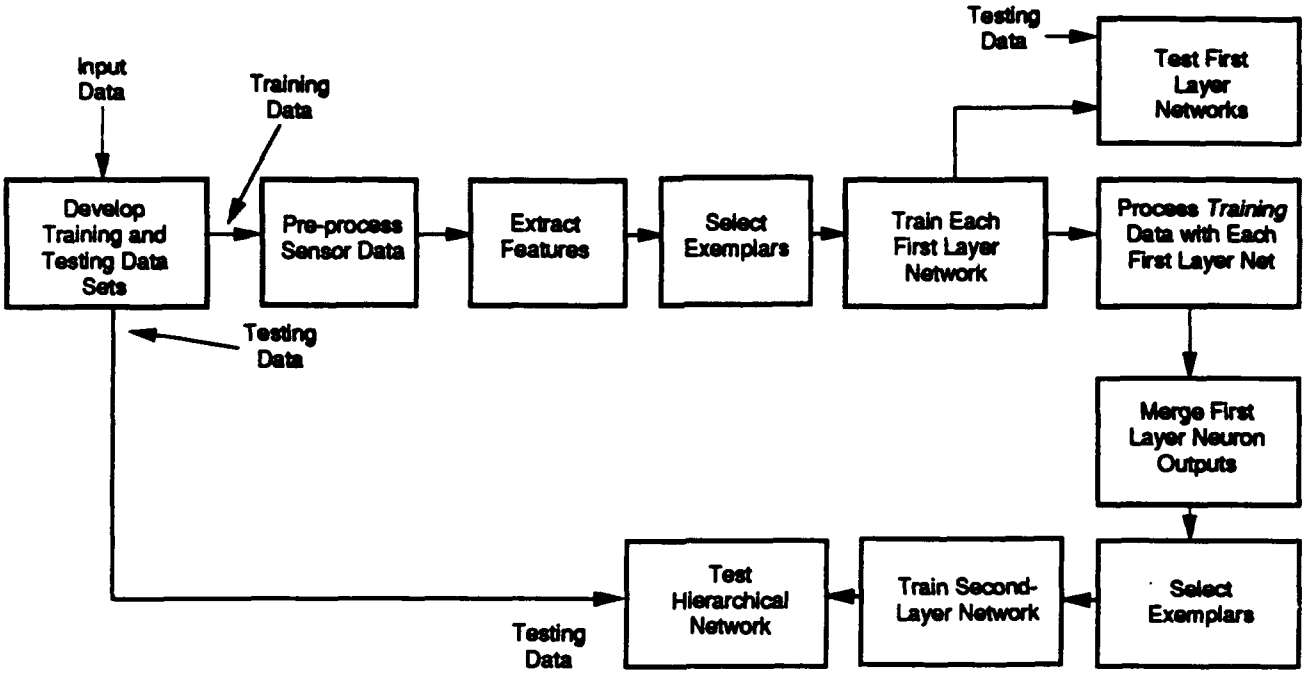


Figure 3. Hierarchical Neural Network Training and Test Flow Diagram

technique does not handle extremely short-duration events (on the order of the FFT size) or very narrowband events (for reasonable FFT sizes). The STFT feature is found as follows. Let $x(t)$ represent the input data set. Let $X_n(f)$ represent the FFT of the n-th segment of data. That is,

$$X_n(f) = \frac{1}{N} \sum_{t=0}^{N-1} w(t)\, x(nT+t)\, e^{-j\frac{2\pi f t}{N}} \tag{1}$$

where typically $T = N/2$ (i.e., "50% overlap") and N is a power of 2. $w(t)$ is a window function, such as the Hamming window. The magnitude squared is taken of the output from FFT processing and then the log is taken to form the FFT feature set characterization of the x(t). The feature input is a set

$$\left\{ \log\left(\left|X_n(f)\right|\right), t_1 \le n \le t_2, f_1 \le f \le f_2 \right\}. \tag{2}$$

This set is used to form the feature space for any particular time/frequency segment of the input data.

- **The Adaptive Optimal Kernel Time-Frequency Representation (AOK TFR):** An instantaneous time-frequency representation (TFR) gives a high-resolution characterization of the data in time, as well as FFT resolutions in frequency for signals of interest. The particular TFR that ORINCON used for Phase I was the

adaptive optimal kernel (AOK) TFR developed by Baraniuk and Jones[6,7]. The AOK TFR is a transformation of Cohen's class, which uses a radially Gaussian *signal-dependent* kernel that changes shape to optimally smooth the distribution.

The optimal kernel, Φ, for a signal is defined as the solution to the following optimization problem:

$$\max_{\Phi} \int_0^{2\pi}\int_0^{\infty} \left| A(r,\psi)\ \Phi(r,\psi) \right|^2 r dr d\psi \tag{3}$$

subject to

$$\Phi(r,\psi) = e^{-\frac{r^2}{2\sigma^2(\psi)}} \tag{4}$$

$$\frac{1}{2\pi}\int_0^{2\pi}\int_0^{\infty} \left| \Phi(r,\psi) \right|^2 r dr d\psi \le \alpha, \quad \alpha \ge 0 \tag{5}$$

$A(r,\Psi)$ is the ambiguity function (AF) of the signal in polar coordinates. $\sigma^2(\Psi)$ measures the spread of the smoothing kernel as a function of angle. Once the optimal kernel is computed, the TFR is given by

$$P(t,\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A(\theta,\tau)\ \Phi(\theta,\tau)\ e^{-j\theta t - j\tau\omega} d\theta d\tau \tag{6}$$

The representation is good for characterizing short-duration and nonstationary events. The AOK TFR is computationally expensive. As with the STFT feature extractor, a time sequence of the AOK TFRs form the input retina.

- **Prony's Model Method**: Prony's model method assumes the signals of interest are modeled by a sum of damped sinusoids. The model is well suited for characterizing impulsive types of events. The Prony model is of the form:

$$x[t] = \sum_{k=1}^{p} A_k \exp\left[(\alpha_k + j2\pi f_k)(t-1) + j\theta_k\right] \tag{7}$$

where $x[t]$ is the observed time series data, p is the model order, A_k is the amplitude of the k-th coefficient, α_k is the corresponding damping term, f_k is the center frequency, and θ_k is the initial phase. The parameters of the model can be estimated using least squares techniques[8]. The spectral estimate is found by Fourier transforming the model defined in equation (7). Since the Prony method assumes that the signals of interest are impulsive, the model found and the spectra derived from the model give

[6] R.G. Baraniuk and D.L. Jones, "A Radially Gaussian, Signal-Dependent Time-Frequency Representation," *ICASSP '91*, Toronto, May 1991.

[7] R. Baraniuk, D. Jones, T. Brotherton, and S.L. Marple, "Application of Adaptive Time-frequency Representations to Underwater Acoustic Signal Processing," *25th Asilomar Conference on Signals, Systems & Computers, vol. 2*, November 1991.

[8] S. L. Marple, *Digital Spectral Analysis with Applications*, Prentice-Hall, 1987.

higher-resolution spectral estimates for impulsive events and sequences of impulsive events than found with standard FFT processing. The Prony model feature set is:

$$\left\{ \log\left(\left|X_n^{\text{Prony}}(f)\right|\right), t_1 \le n \le t_2, f_1 \le f \le f_2 \right\}, \tag{8}$$

where $X_n^{\text{Prony}}(f)$ is the spectral estimate found as described above.

The model is relatively easy to estimate, so it is computationally efficient when compared to the instantaneous time-frequency representations, but still slower than the STFT. Once the exponential parameters have been estimated by the Prony procedure, spectral plots are created by taking the Fourier transform of the resulting model. This technique, along with other parametric modeling techniques, has better time and frequency resolution properties than the STFT.

- **Wavelet Transform:** Wavelet analysis, also known as time-scale analysis, is essentially a method of "constant Q" filtering[9]. The transform is implemented using a set of band pass filters whose bandwidths are proportional to their center frequencies. Thus, very narrow bandwidth filters (with long time durations) were applied at lower frequencies, while wider bandwidth (with very short time durations) filters were applied at higher frequencies. Wavelet transforms are especially useful when short-duration transient events are superimposed upon long-duration, low-frequency components.

  The wavelet transform of a function $f(x)$ represents a decomposition of the function in terms of dilated and shifted versions of an analyzing wavelet function $\psi(x)$. The transform is linear, energy preserving, and invertible so for every function $f(x)$, there is a unique, continuous 2-D transform $Wf(s,u)$. The wavelet transform can also be regarded as a time-frequency representation of the signal, in which the parameter $s > 0$ corresponds to frequency and the parameter $u \in R$ corresponds to temporal shift. The wavelet transform has many interesting properties that make it particularly well suited as a signal representation.

  The wavelet transform can be viewed as a time-frequency map of the signal in which the frequency information is generated by a bank of proportional bandwidth filters. This entire map can be regarded as a single feature of the signal. Because of the inverse relationship between bandwidth and temporal support, the wavelet time-frequency representation automatically provides greater temporal resolution for high-frequency signal components. This is often very useful when analyzing transient or highly nonstationary phenomena.

  The wavelet features presented herein were generated using an analytic Gabor wavelet. The wavelet is defined in the frequency domain by the following equation

$$H(\omega) = \begin{cases} \dfrac{1}{2i}\left(e^{-\frac{\sigma^2(\omega-\omega_0)^2}{2}} - e^{-\frac{\sigma^2(\omega+\omega_0)^2}{2}} \right) & \text{if } \omega \ge 0 \\ 0 & \text{if } \omega < 0 \end{cases}, \tag{9}$$

  where $\omega_0 = 2\pi$ and $\sigma = 6.1182$. This choice of parameters results in a time-frequency map in which the bandwidth of each frequency bin is approximately 1/16th of an octave.

[9] O. Rioul and M. Vetterli, "Wavelets and Signal Processing," *IEEE Signal Processing Magazine*, October, 1991.

2.3 Neural Network Classifier

The fundamental building blocks of the fault detection and classification system are the individual neural networks. There are a multitude of classifiers that can be loosely regarded as "neural" architectures. During training, neural networks learn how to interpret the input features to solve the detection/classification problem. Essentially, neural nets can be viewed as nonlinear matched filters. They simultaneously detect and classify signals of interest when they appear in their input retinas. For the Phase I effort, we used a hierarchical neural net approach as shown in Figure 2. Each neural net in the system is a multilayer perceptron neural network trained using the back-propagation algorithm. Multilayer perceptron nets have been used extensively and successfully at ORINCON for various detection and classification problems. Specifically, all of the neural networks used in the Phase I systems were three-layer perceptrons with 20 hidden nodes in each of the two hidden layers. The neural networks were trained such that their mean-squared output activation errors were between 0.009 and 0.026. The three-layer perceptron architecture is shown in Figure 4. For solving the Phase I problem, the multilayer perceptron (MLP) approach is fine since this architecture has been shown to implement a Bayesian classifier[10]. For novelty detection (i.e., detection and classification of new, unknown events), the MLP/Bayesian approach may not be adequate[11]. For Phase II, alternative neural net architectures will be considered.

3.0 REAL DATA PROCESSING RESULTS

3.1 Hollins Helicopter Gearbox Data

The Hollins data set provided by ONR is two-channel vibrational data from two separate accelerometer sensors (sensor tracks 5 and 6) recorded from the gearbox in the tail of a TH-1L helicopter. The data provided included examples of normal operation as well as examples with the following five seeded fault conditions: Bearing, Inner Race; Bearing, Rolling Element; Bearing, Outer Race; Gear, Spall; and Gear, 1/2 Tooth Cut.

The initial system developed for the helicopter data was a single-channel, multifeature approach. That is, only one channel (sensor track 5) was used in the processing, but data from this channel was processed using all four feature extractors discussed in Section 2.2[12]. Figure 5 shows the

---

[10] M. Richard and R. Lippmann, "Neural Network Classifiers Estimate Bayesian *a posteriori* Probabilities," *Neural Computation*, 3, 461-483, 1991.

[11] A. Leonard and M.A. Kramer, "Radial Basis Function Networks for Classifying Process Faults," *IEEE Control Systems Magazine*, April 1991.

[12] T. Brotherton, T. Pollard, and D. Jones, "Application of Time-Frequency and Time-Scale Representations to Fault Detection and Classification," *Proc. of the International Symposium on Time-Frequency and Time-Scale Analysis*, Victoria, B.C., Canada, October, 1992.

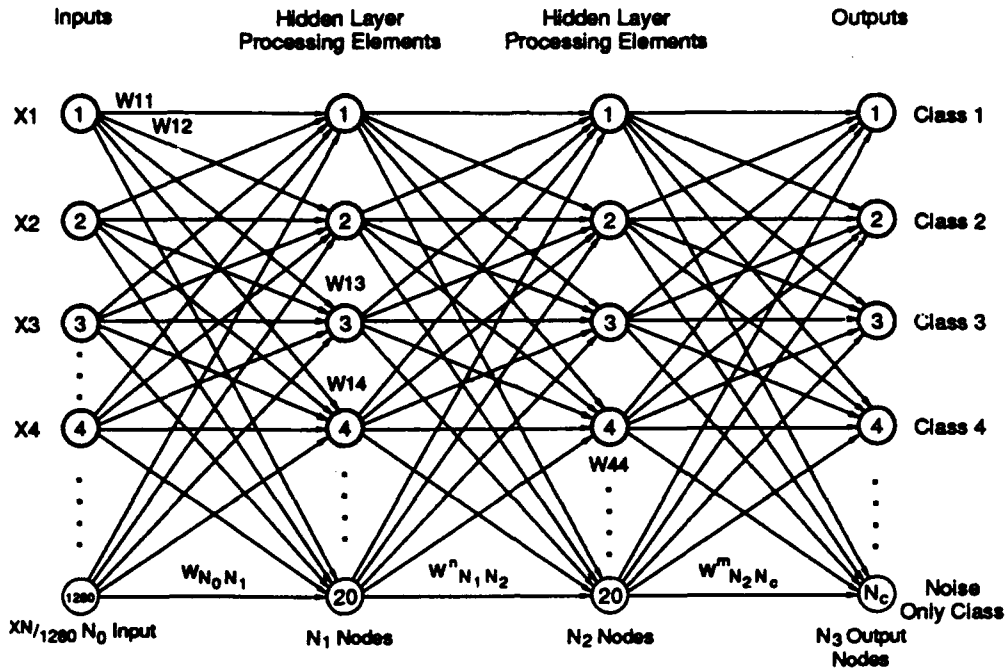


Figure 4. Three-Layered Perceptron Neural Network Used by ORINCON for Feature Space Detection and Classification

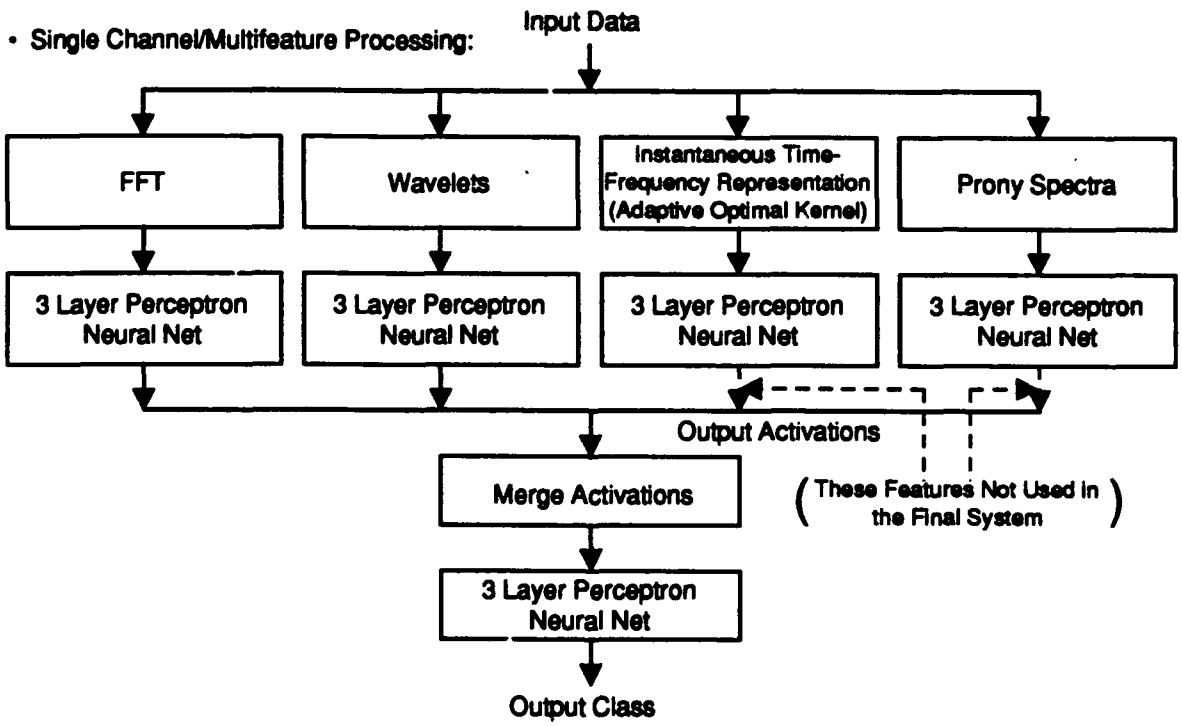


Figure 5. Single-Channel, Multifeature Processing of Hollins Data Set

single-channel, multifeature architecture. The first-and second-layer nets were trained using the methodology outlined in Section 2.0.

In Figure 5, each of the feature extractors is one of the alternative spectral estimates described in Section 2.0. The input to the first-layer networks is a two-dimensional array, in time and frequency, of spectral estimates. Let $X_n^M(f)$ denote a spectral estimate found using technique M at time scan n and frequency f. $M \in$ {FFT, wavelet, Prony's method, or the AOK time/frequency representation}. Then the retina input to each of the first layer networks is

$$\left\{\log\left|X_n^M(f), t_1 \le n \le t_2, f_1 \le f \le f_2\right|\right\}. \tag{10}$$

Figure 6 is an example of the system's operation. The input data is a cut that includes the Bearing, Outer Race fault. Figure 6 shows five separate windows, with one window for each of the first-layer feature extractors/neural nets, and one window for the second-layer fusion net. Each of the windows has two sections. Figure 7 is a cartoon showing the details for the first-layer feature/neural net windows in Figure 6. For the first-layer feature extracts/neural nets the section on the left of the displays shows the time series of features extracted from the data. That is, each row on the left is a separate spectral estimate of the data using the indicated feature extractor, with time running on the y-axis and frequency on the x-axis. These feature vectors are accumulated into multirow, two-dimensional retinas, described above, which are used as the input to the first-layer neural nets. With each new spectral estimate, the input retina is updated and fed to the corresponding neural network.

The right-hand portion of each window represents the neural net's output for each retina input. Each column represents the activation level corresponding to each class of interest. Column 6 (the rightmost column) represents the normal condition, i.e., no faults. The neural net outputs are temperature encoded, with white indicating the highest activation level (high probability of class membership), and black representing the lowest activation level (low probability of class membership). The parameter settings used for each of the feature extractors and hierarchical neural networks are shown in Table 1.

The Bearing, Outer Race fault was designated as class number 3, meaning that ideally the net activations for column 3 would be white, with all other columns black. However, one can see that all of the first-layer networks have trouble with the distinction between classes 2 and 3.

The Prony and AOK neural networks show a great deal of misclassification. However, the fusion net demonstrates ideal performance. The AOK and Prony neural networks had consistently poorer performance when compared to the FFT and wavelet neural networks.

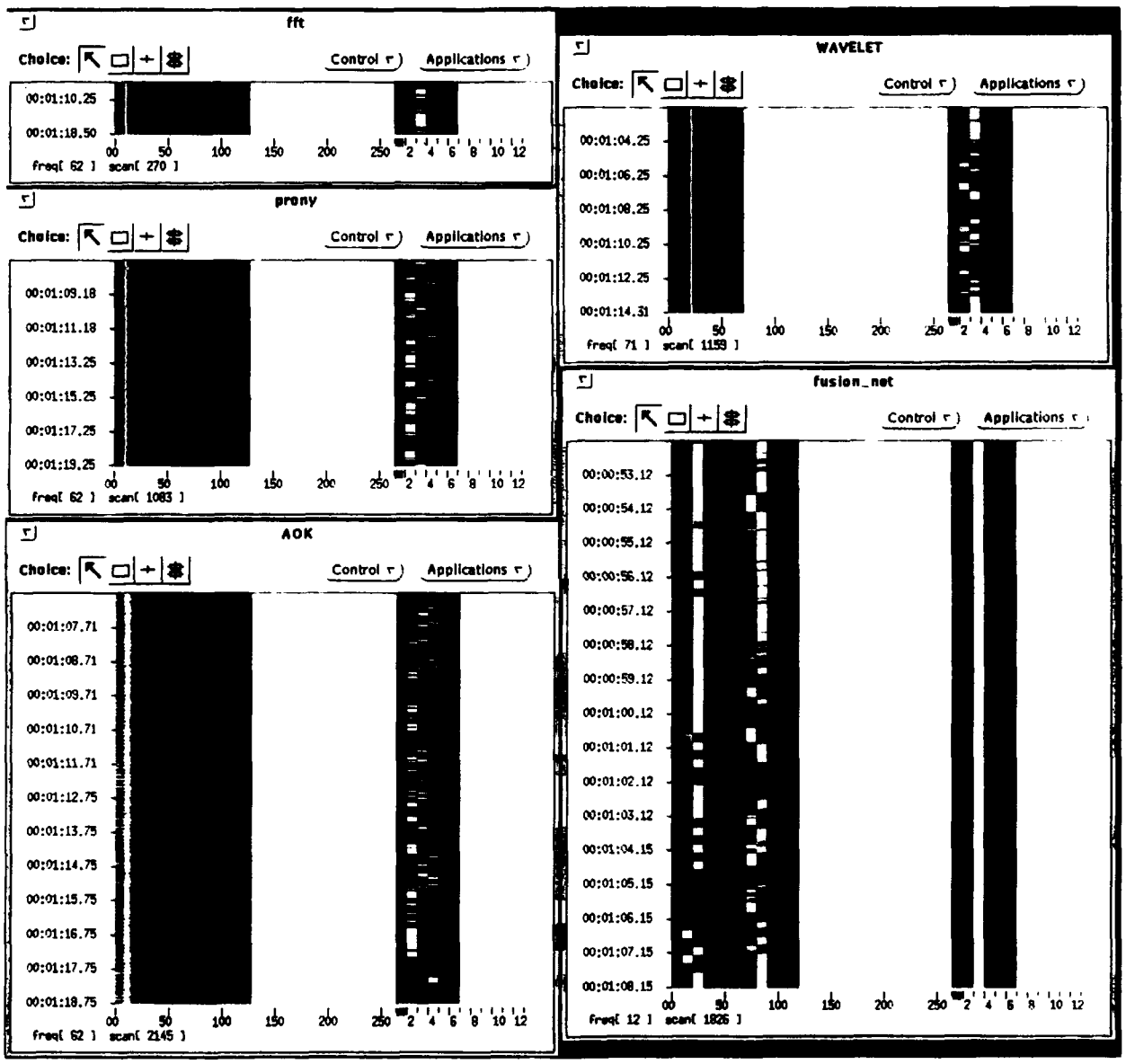


Figure 6. Bearing, Outer Race Fault in Hollins Multifeature System

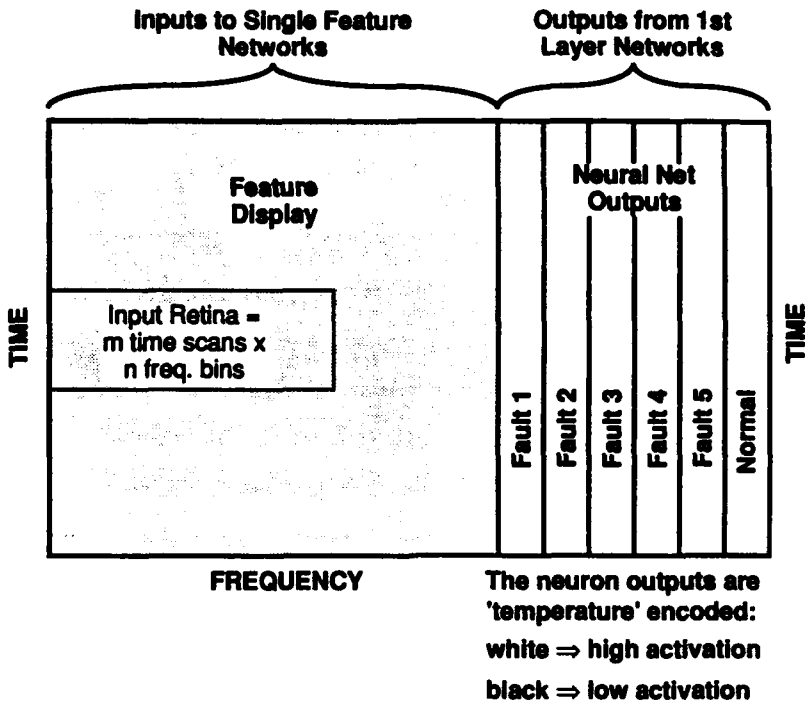


Figure 7. Inputs and Outputs for the First-Layer Nets

Figure 8 shows an expanded view of the AOK feature extractor applied to the Gear, 1/2 Tooth Cut data.

The problem of the AOK and Prony methods probably has two causes. First, no attempt was made to synchronize the data for each rotation (no sync, or tachometer, signal was provided). Thus, when input retinas were selected for these feature extractors, the gears were in a random rotational state, thus increasing retina-to-retina variance for the very-high-time-resolution AOK and Prony feature extractors. In the future, synchronizing the high-resolution time-frequency feature extractors with a sync signal may improve their classification performance.

In addition, the neural network can only have good classification performance if the testing data "looks" similar to the training data. If there is variation between the training and testing data, the neural net will have degraded performance. It is probable that the impulsive transients seen with a fault condition have a great deal of variance from rotation to rotation (refer to the periodic transients indicated in Figure 8). Because of the high time-resolution properties of the Prony and AOK feature extractors, these transient variations are characterized quite well, thus producing input retinas for the neural nets with a high degree of variance, which degraded classification performance. In an interesting twist, the lower-resolution (in time) feature extractors, the FFT and the wavelet, always had superior performance. The inherent time averaging and smoothing of these feature extractors helped to accentuate only the stationary fault features necessary for

Table 1. Single-Channel Processing Parameters

FFT	WAVELET	AOK TFR	PRONY'S METHOD	FUSION NET
Window Size = 256	Analyzing Wavelet = Gabor	Window Size = 128	Window Size = 64	Retina: 200 Time Scans x 12 Inputs
FFT Size = 256	No. of Scale Factors = 71	FFT Size = 256	Model Order = 12	
Window = Hamming	Low Bin Center Freq. = 768 Hz	Kernel: Volume = 2.5	FFT Size = 256	
Samples Skipped = 128	High Bin Center Freq. = 16642.56	No. Grad. Steps = 6	Samples Skipped = 32	
Retina: 5 Time Scans x 60 Frequency Bins (bins 2 - 62)	Samples Skipped = 32	Samples Skipped = 16	Retina: 10 Time Scans x 60 Frequency Bins (bins 2 - 62)	
	Retina: 10 Time Scans 71 Frequency Bins	Retina: 20 Time Scans x 60 Frequency Bins (bins 2 - 62)		

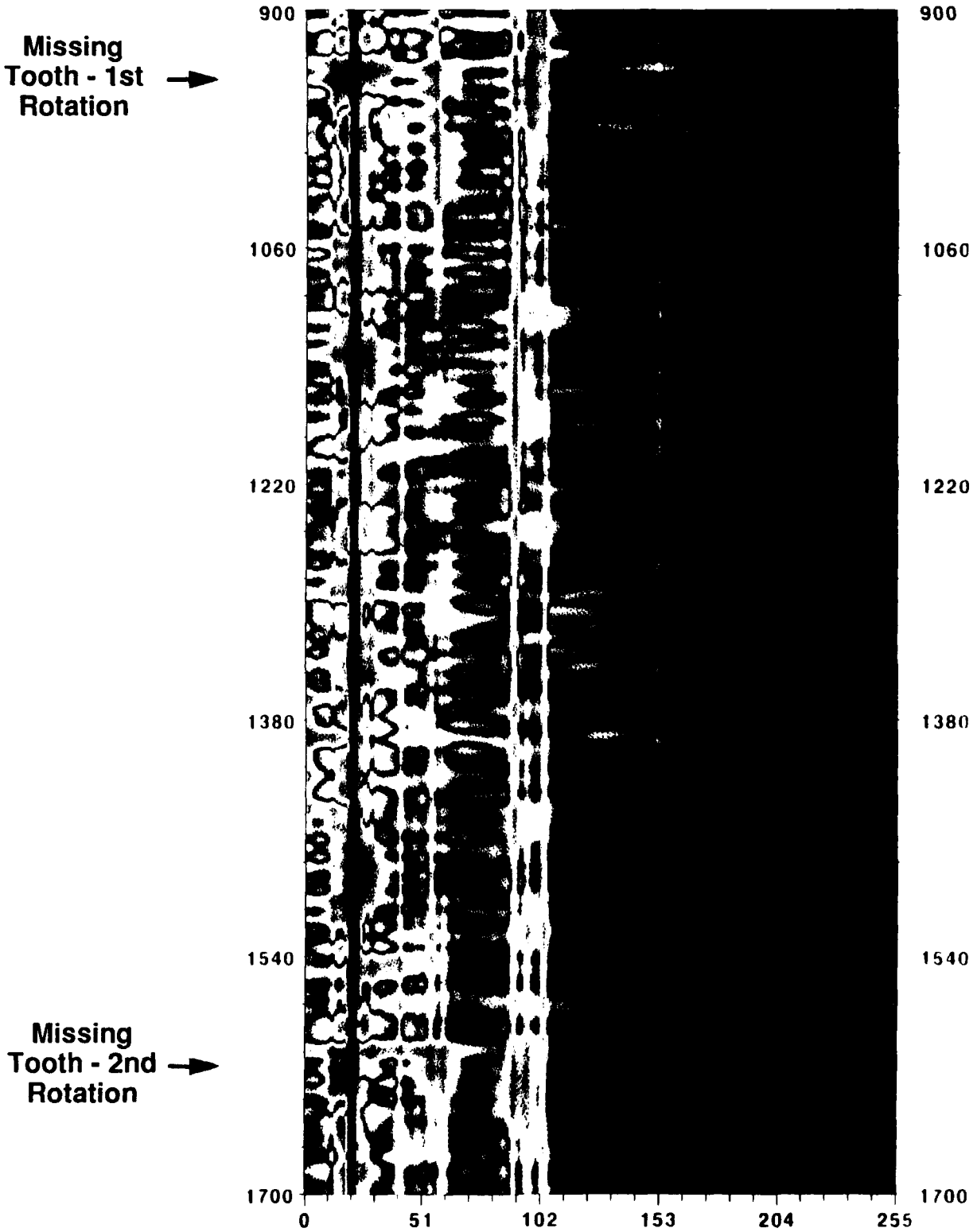


Figure 8. AOK TFR Applied to Hollins Cut-Tooth Data

classification. Thus, in the single-channel, multifeature system of Figure 5, only the FFT and wavelet outputs were merged for the fusion net. The connections for the Prony and AOK net outputs were, in effect, cut.

Returning to Figure 6, the fusion_net window shows the merged (appended) FFT and wavelet activation vectors on the left, and the fusion net outputs on the right. Figure 9 is a depiction of the input and output sections of the fusion net window of Figure 6. The output activations from the first-layer nets have been replicated for the purposes of system synchronization (FFT net activations by a factor of 8, and wavelet net activations by a factor of 2). Unlike the first-layer nets, the fusion net performance is essentially perfect, with column 3 solidly lit, no false alarms, and no misclassifications. The fusion net has successfully fused and integrated the information from the FFT and wavelet first-layer nets to improve the system performance.

The results shown in Figure 6 prompted the development of the system shown in Figure 10. This is the final system used for quantitative testing on the Hollins data set. For this system, only FFT (1,024-point) processing was used for the first-layer nets. The system shown in Figure 10 is a multichannel system in which each sensor channel is processed with a 1,024-point FFT, and the frequency range is partitioned into low and high frequencies for input into four separate first-layer nets.

In this case, it was found that the networks associated with channel 2 (sensor track 6) did not improve the classification performance. The reason for this is unclear. Thus, only the two channel 1 nets (low and high frequencies) were merged for input to the second-layer fusion network. Unlike the fusion net of Figure 6, the merged inputs to the fusion net of Figure 10 have been rearranged, with neurons from the first-layer nets corresponding to the same class grouped into neighboring positions in the merged fusion vector as shown in the insert on Figure 10. This regrouping makes no difference in the performance of the neural network, but it does aid the human observer. All of the processing examples in the remainder of this proposal used this regrouping methodology. Table 2 shows the processing parameters used for this system shown in Figure 10.

Figures 11 and 12 show the response of the system to a cut containing the Bearing Inner Race fault (class 1 - Figure 11) and the Bearing Roller Element fault (class 2 - Figure 12). For these examples, the input data file was not used in the system training. Thus, the system had never been exposed to the data shown. The partitioning of training and testing data is discussed more fully below. The input retinas used are shown in the channel 1 and fusion net windows. Like the preceding example, both channel 1 nets do a relatively good job of making the fault classification, but the fusion net, after merging the two channel 1 nets, provides essentially perfect performance.

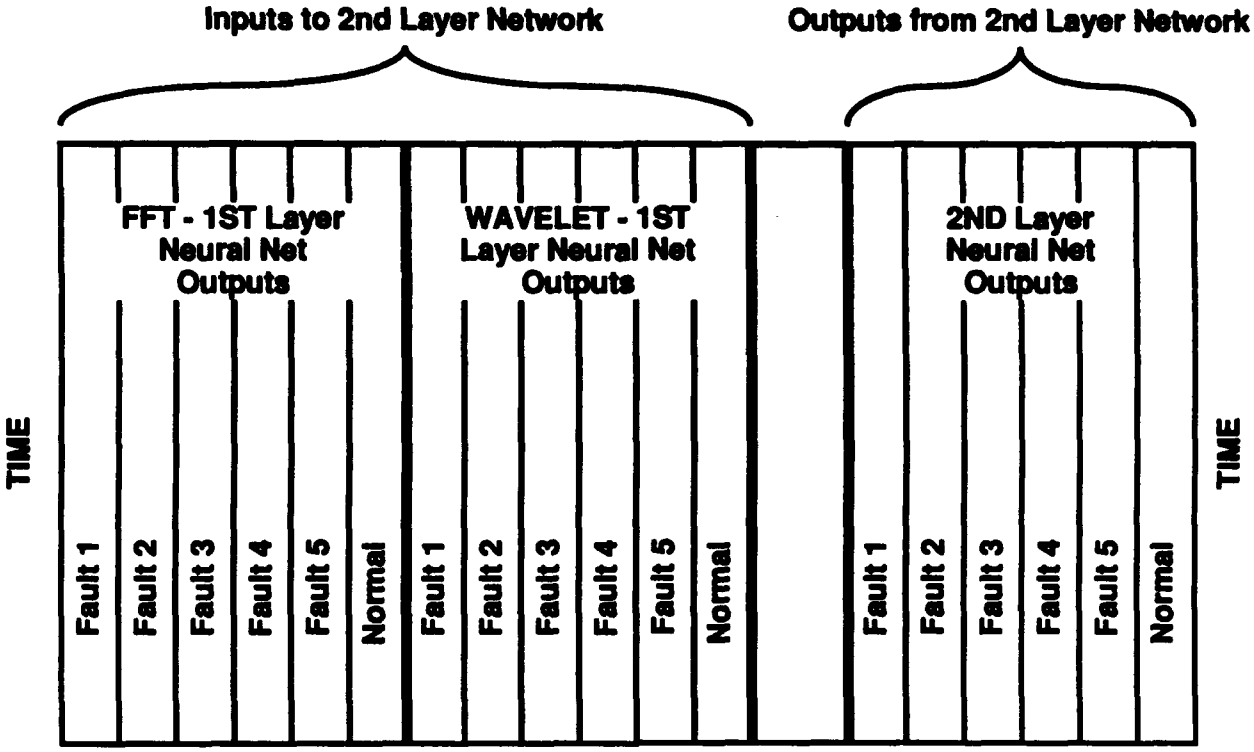


Figure 9. Inputs and Outputs for the Second-Layer Nets

Table 2. "Multichannel" Processing Parameters

- F_S = 48 KHz
- FFT = 1024 pts
- Hamming Windowed
  50% Overlap

⇒ Low Frequency Retina
  Bins 1-225 (50-10546 Hz)
  x 10 Time Scans

⇒ High Frequency Retina
  Bins 220-469 (10312-22000 Hz)
  x 10 Time Scans

- Fusion Net Retina
  12 Classes
  x 100 Time Scans

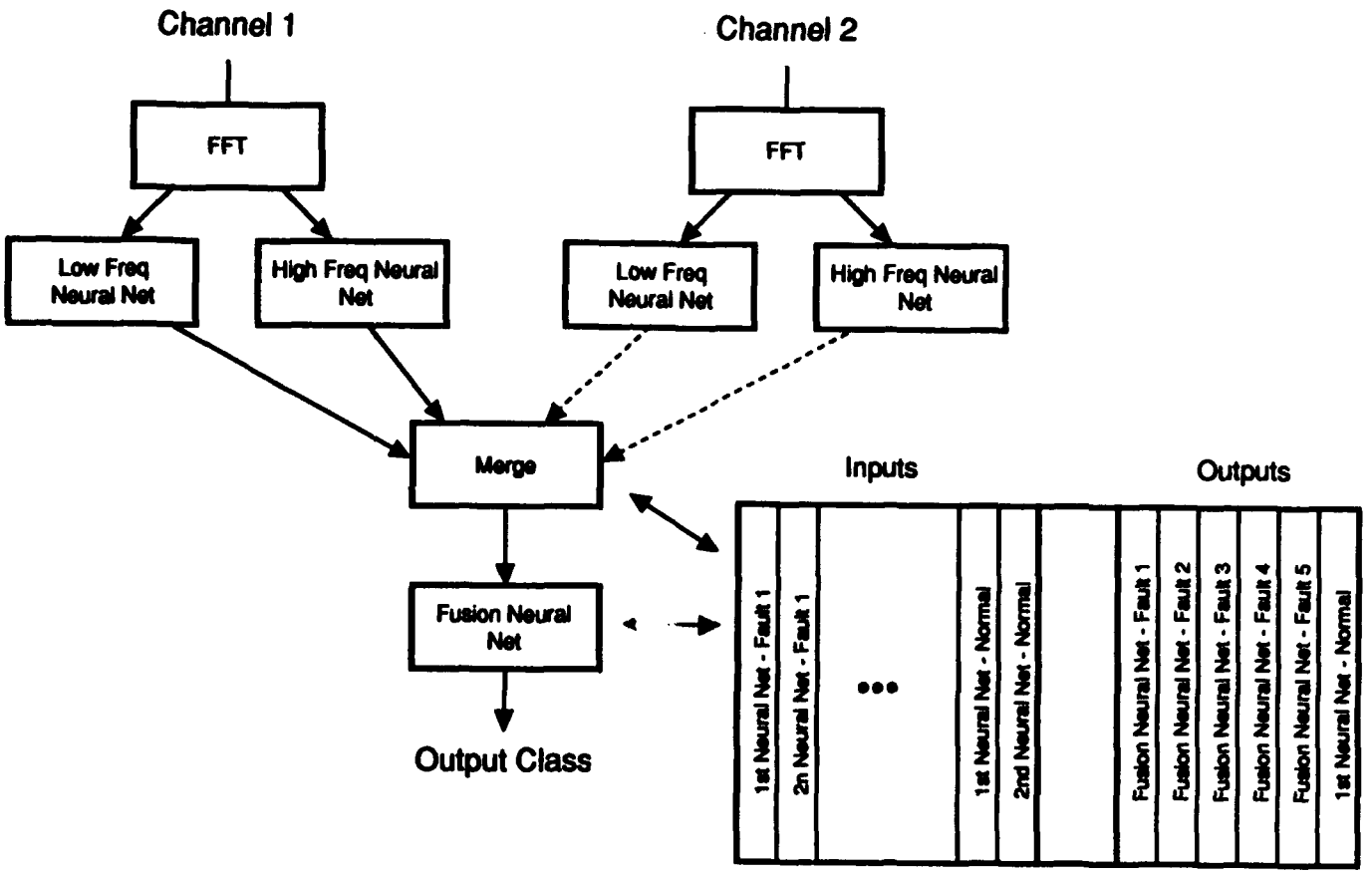


Figure 10. "Multichannel" Processing Flow Diagram for the Hollin's Data Set Full

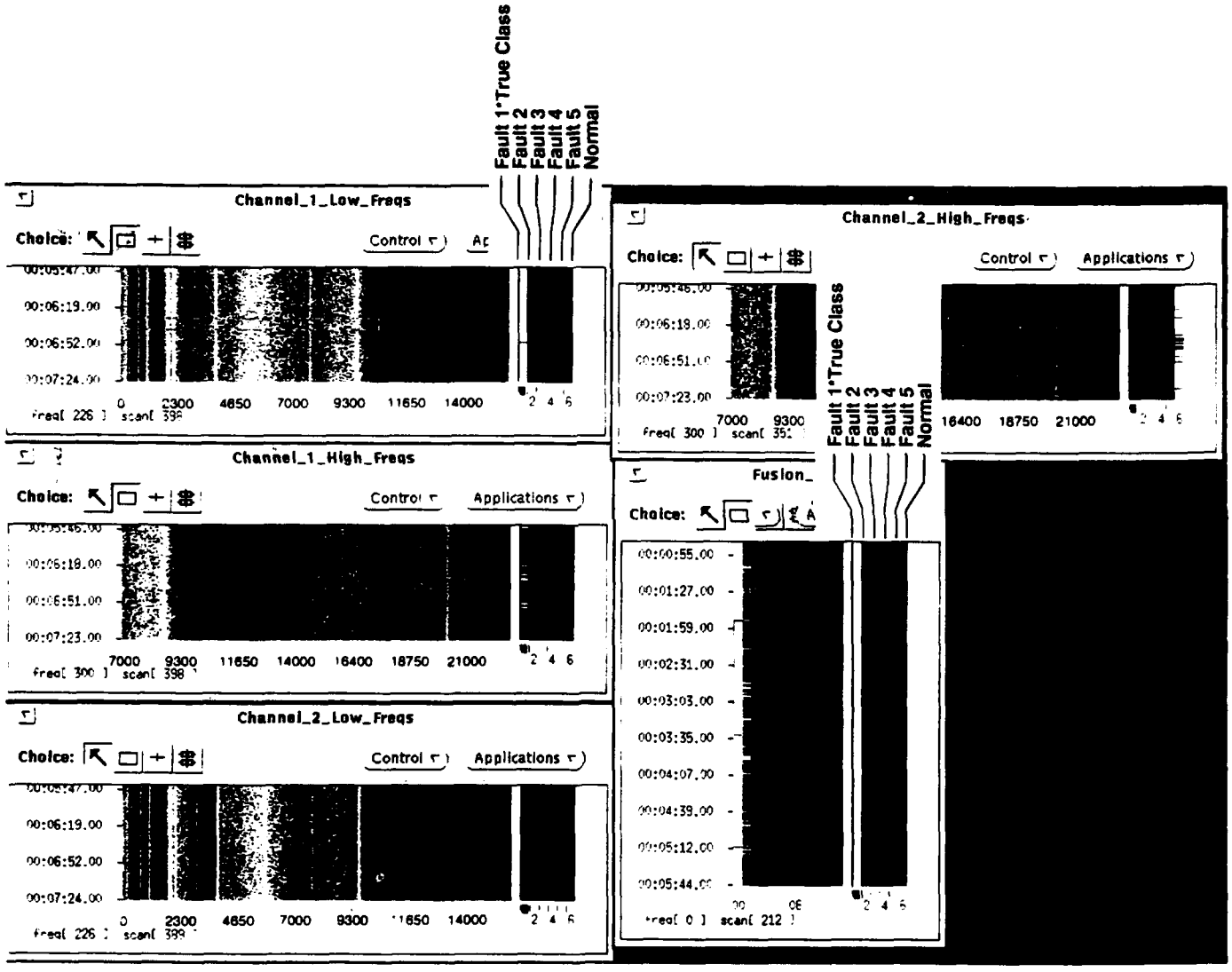


Figure 11. Bearing Inner Race Fault Example

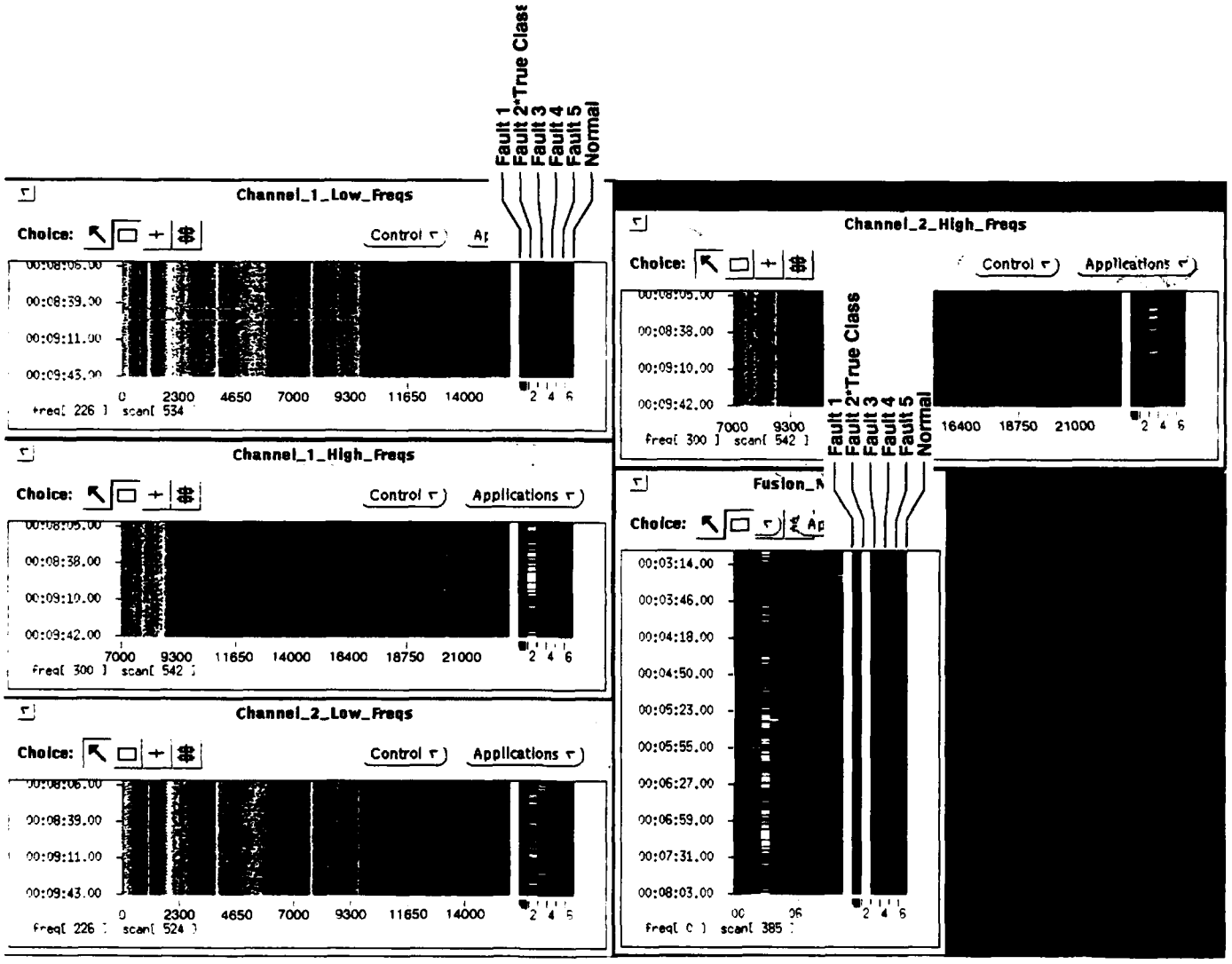


Figure 12. Bearing Rolling Element Fault Condition

SRI INTERNATIONAL CORPORATION

Quantitative performance measurements were made using test data. The files used were 60 seconds in length, and training and testing data were partitioned as follows. When more than one file was provided for a given fault class, a subset of files were used for training, and the remaining files were used for testing. Often, only one file was provided for a given class. In these cases, the first portion of the file was used for training, and the second portion of the file was used for testing. Thus, training data was selected from the first portion (15 seconds) of training files, and testing data included the last portion (45 seconds) of training files and the entire duration of the separate testing files not used in training. Because of the relative stationarity of the data within a given file, it is the performance on the separate testing files, which the system had never been exposed to, that provides a true measure of system performance.

A decision threshold was defined as the difference between the highest fault class activation level and the normal class activation level. A fault detection was declared if the difference between the highest fault class activation and the normal class activation exceeded a predefined threshold. In general, this threshold value is set to zero, and the classification decision simplifies to that class with the highest activation. For all of the test results presented here, the threshold value was equal to zero, and the highest activation was the declared class.

Table 3 shows a confusion matrix for the system of Figure 10. The input fault class is indicated on the horizontal axis, and the resulting system output is indicated on the vertical axis. Each element of the matrix indicates the number of times that the system declared the output class and the number of decision opportunities for each class. Ideally, the confusion matrix should be diagonal, indicating 100 percent correct classification. However, Table 3 shows that the system had some confusion between fault classes 2 (Bearing, Rolling Element) and 3 (Bearing, Outer Race) when class 3 was input to the system. Further investigation and examination of the data showed that one of the class 3 files, cut 8, has a spectrum similar to the class 2 data, thus causing the confusion. Table 3 represents the testing when the system was not trained using cut 8. Note that although there is some fault classification confusion in Table 3, there are no missed fault detections and no false alarms.

To improve the system performance, cut 8 was subsequently included in the training set, and testing was performed again. After being trained with cut 8, the system confusion matrix was diagonal. Even after training with cut 8, there were still two files (one normal, and one Bearing, Inner Race fault as seen in Figure 11) used in the testing that had not been used in the training. Table 4 summarizes these results. Because there were no false alarms throughout the testing, the probability of false alarm, P_{FA}, is upper bounded by the reciprocal of the number of normal class decisions.

Table 3. Hollins Data Confusion Matrix Trained Without Cut 8

System Output \ Truth	Fault 1	Fault 2	Fault 3	Fault 4	Fault 5	Normal
Fault 1	8550 8550 =100%					
Fault 2		3664 3664 =100%	4886 8550 =57%			
Fault 3			3664 8550 =43%			
Fault 4				3664 3664 =100%		
Fault 5					8550 8550 =100%	
Normal						12214 12214 = 100%

Table 4. Hollins Data Performance Summary

Data	Threshold ①	P(d)	P(fa) ②	Confusion Matrix Summary
Trained with Cut 8 Test All Data	0	1	$<8\times10^{-5}$	Diagonal
Trained without Cut 8 Test All Data	0	1	$<8\times10^{-5}$	Diagonal except 57% of Class 3 Misclassified as Class 2

① Nominal Threshold Setting = 0 (i.e., Max Fault Neuron = Normal Neuron).

② P(fa) Estimates on Per Scan Basis. When NO False Alarms in Data P(fa) = 1/(total # of normal condition scans).

3.2 Fire Pump Data

The second set of data processed was six-channel vibrational data from multiple fire pumps. The system used to process this data is shown in Figure 13. Table 5 shows the processing parameters used. In this architecture, each channel of data is processed with 1,024-point FFTs. Each channel has its own neural network, and as before, the first-layer network outputs are merged in a fusion net for the final fault classification. The fire pump data had four fault conditions, as well as the normal condition (five classes).

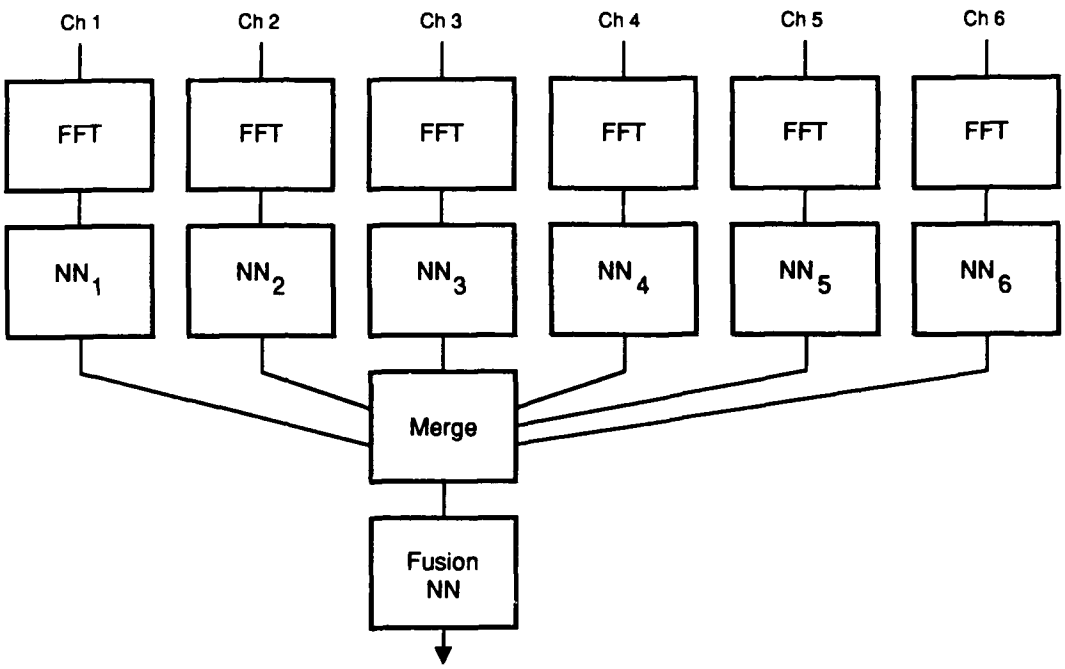


Figure 13. Fire Pump and Condenser Pump Multichannel Processing Flow Diagram

Table 5. Fire Pump Multichannel Processing Parameters

F_S	=	50 KHz
FFT	=	1024 pts Hamming Windowed 50% Overlap
Retina (Same for Each Channel) Bins 1-200 (50-9765 Hz) x 10 Time Scans		
Fusion Net Retina 30 (6 Channels x 5 Classes/Channel) x 100 Time Scans		

Two examples of the processing are shown in Figures 14 and 15. Figure 14 shows the system response to a cut containing fault class 1. This file was not used in the system training. The faint boxes in the channel 4 and fusion net windows indicate the retinas used for the respective neural networks. One can see that channels 4 and 6 do an excellent job of classifying the fault, and channel 5 does a partial job of correct classification. Channels 1, 2, and 3 do not classify the fault condition correctly. The fusion net, however, successfully arbitrates and integrates the first-layer information to achieve perfect performance.

Figure 15 shows the system response to one of the normal condition files. Again, the system was not trained with this cut. In this example, ideal performance would be for column 5 to be white, with all other (fault) columns black. One can see that all of the channels do relatively well, except channel 6, which does very poorly, false alarming consistently on fault class 4. Again, the fusion net gives perfect performance.

The above examples demonstrate the necessity for multichannel measurements and fusion. The data is variable enough that there is no guarantee, for a given fault condition, that the test data (Figures 14 and 15) will look like the training data for every channel. If one were dependent on only one channel for fault classification, the resulting performance would be severely degraded. By making use of all of the channels, there is a higher probability that a subset of the channels will make the proper fault classification, thus allowing the fusion architecture to make the final correct decision.

Table 6 summarizes the fire pump system performance. Again, the data was separated into training and testing portions using the methodology outlined in Section 2.0. For the fire pump data set, the files provided were typically 15 seconds in length. Training data was selected from the first 4.26 seconds of the training files. Test data was taken from the last 10 seconds of the training files, as well as from the entire duration of the subset of testing files not used in training. Table 6 indicates the performance on all of the test data, as well as on only the test data taken from files that had not been used in the training (six out of 16 files). For this data set, there were no false alarms, no missed detections, and no misclassifications. The P_{FA} performance is upper-bounded in the same manner described previously.

3.3 Condenser Pump Data

The condenser pump data set had the same format as the fire pump data, that is, six channels of vibrational data. The architecture used for the condenser pump data processing was the same as that for the fire pump, and was shown in Figure 13. The condenser pump data set had two fault conditions, as well as the normal condition (three classes). The processing parameters used are shown in Table 7.

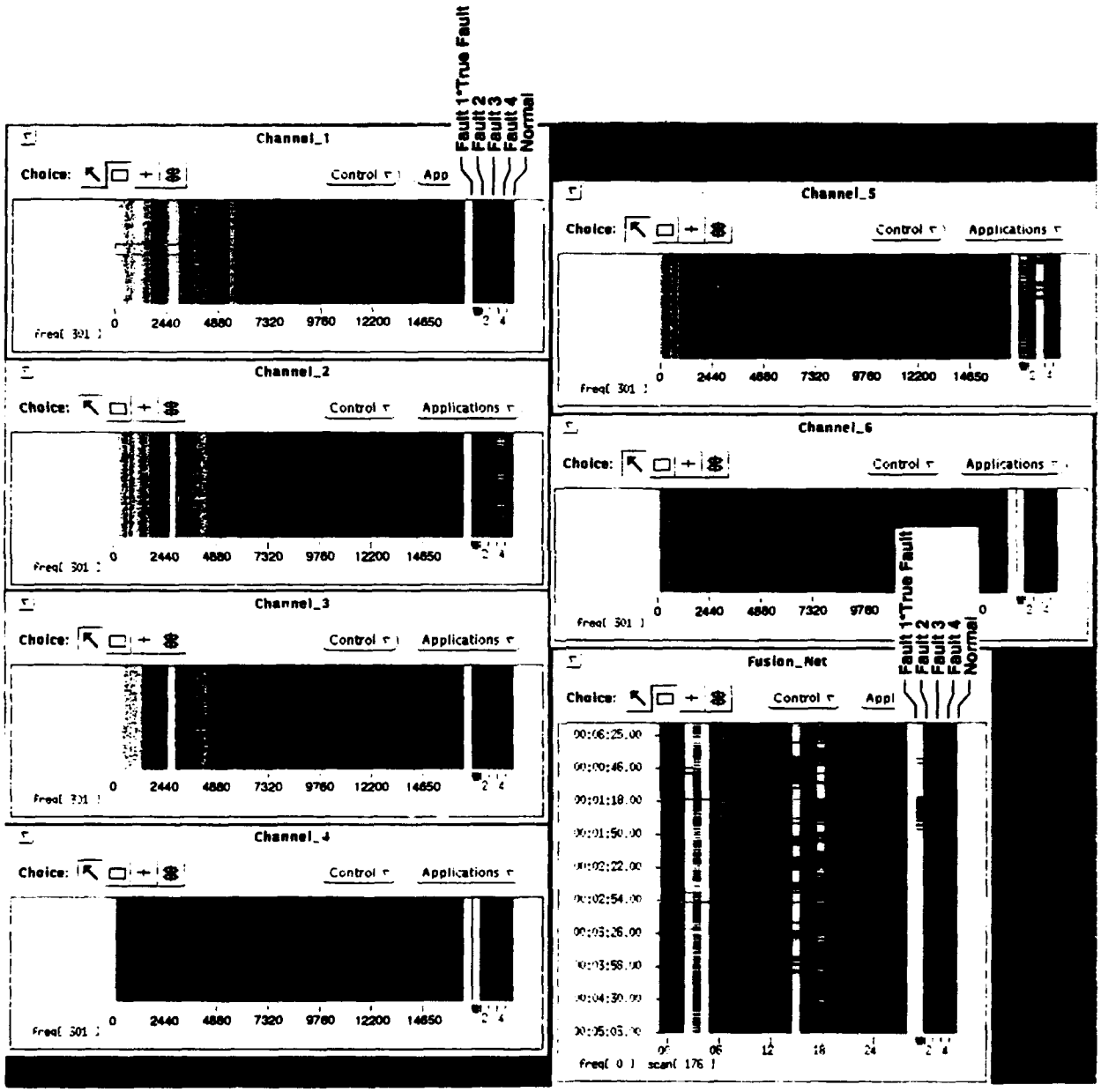


Figure 14. Fire Pump System with Class 1 Fault

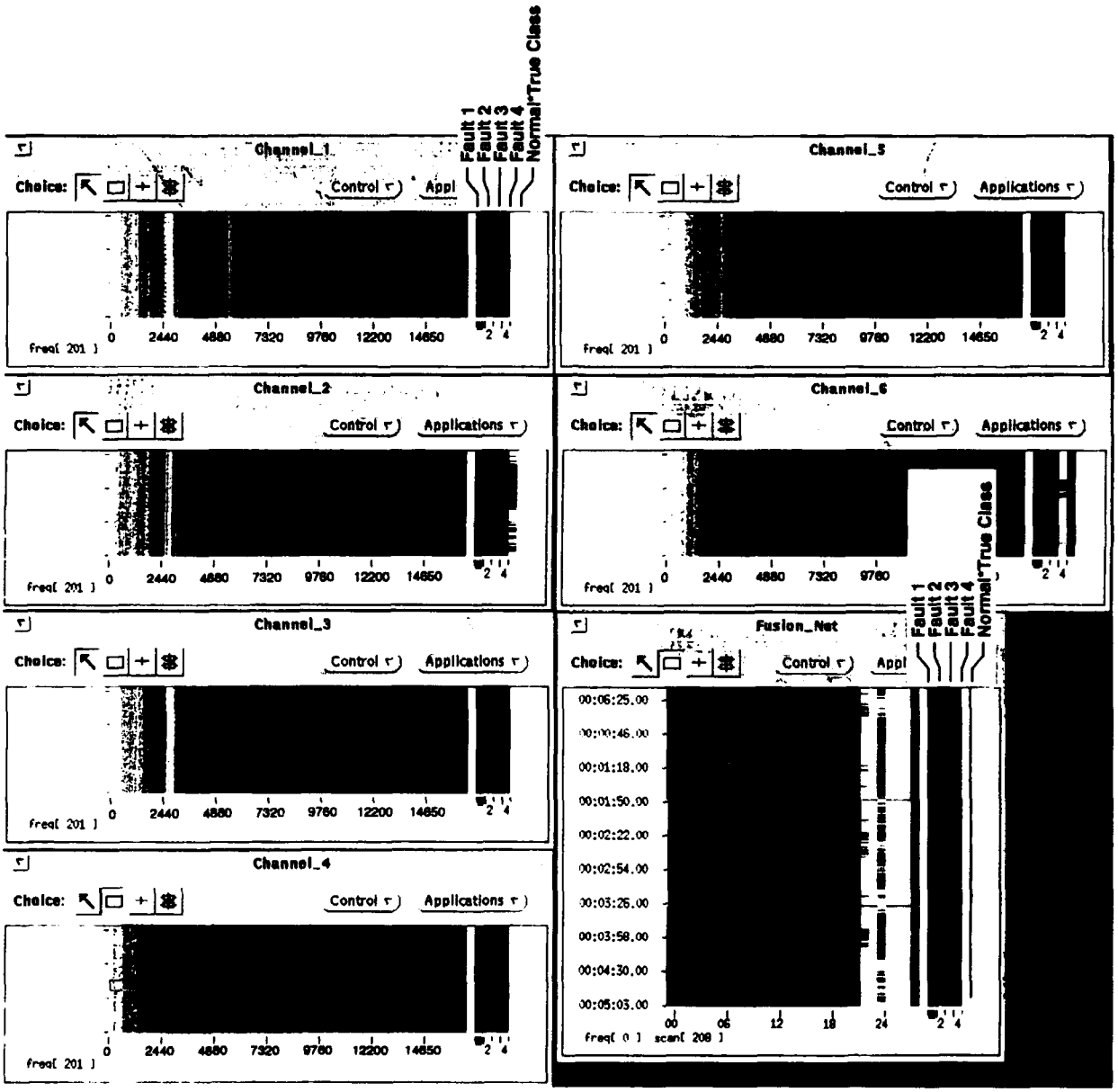


Figure 15. Fire Pump System with Normal Data

Table 6. Fire Pump System Performance Summary

Data	Threshold ①	P(d)	P(fa) ②	Confusion Matrix Summary
Test All Data	0	1	$<8 \times 10^{-5}$	Diagonal
Test on Separate Test Data	0	1	$<1.5 \times 10^{-4}$	Diagonal

① Nominal Threshold Setting = 0 (i.e., Max Fault Neuron = Normal Neuron).

② P(fa) Estimates on Per Scan Basis. When NO False Alarms in Data P(fa) = 1/(total # of normal condition scans).

Table 7. Condenser Pump Processing Parameters

F_S = 50 KHz

FFT = 1024 pts
Hamming Windowed
50% Overlap

Retina (Same for Each Channel)
Bins 1-100 (50-5000 Hz)
x 10 Time Scans

Fusion Net Retina
18 (6 Channels x 3 Classes/Channel)
x 100 Time Scans

Examples of the processing are shown in Figures 16 and 17. Figure 16 shows the system response to a class 2 fault condition. Retina boxes are drawn in the windows for channel 6 and the fusion net. As for the previous examples, the results shown are for independent test data. One can see that channels 4, 5, and 6 do relatively well, while channels 1, 2, and 3 have very poor performance. Again, the fusion net successfully merges the information and produces a perfect classification.

Figure 17 shows the system performance with normal condenser pump data. Again, the system was not trained with this cut. One can see that for this example, all of the channels have marginal performance, with channel 3 completely misclassifying the normal data as a fault (false alarm). The two best channels are 1 and 2, in contrast to the example of Figure 16. Like the fire pump examples above, these two examples indicate the utility of multichannel measurements and data fusion in the hierarchical neural net system. Again, the fusion net provides essentially perfect performance.

A performance summary for the condenser pump data is given in Table 8. This table has the same format as Table 5 (Section 1.4.2). Four of the eight files provided were not used in the training, and their test results are in the lower portion of the table. Again, there were no misclassifications, missed detections, or false alarms during the testing.

4.0 CONCLUSIONS

The primary conclusions to be drawn from the Phase I effort and test results are as follows:

System Flexibility. The generic hierarchical system developed in Phase I was proven to be extremely flexible. The same general architecture has been applied to three very different data sets, with perfect performance for each. The system is capable of extension and expansion, depending upon the number of sensor types, number of channels, and number of feature extractors that prove useful toward solving the problem. Because the hierarchical neural net system is not model based, it can be quickly reconfigured and retrained to accommodate machinery design changes or new fault detection tasks.

Perfect System Performance. The hierarchical systems developed for the three data sets provided perfect performance. That is, with proper training, the systems achieved perfect fault/normal classification for all of the test data run. However, because the test data was limited, more extensive testing and quantitative analysis would need to be undertaken to produce more reliable statistical performance estimates.

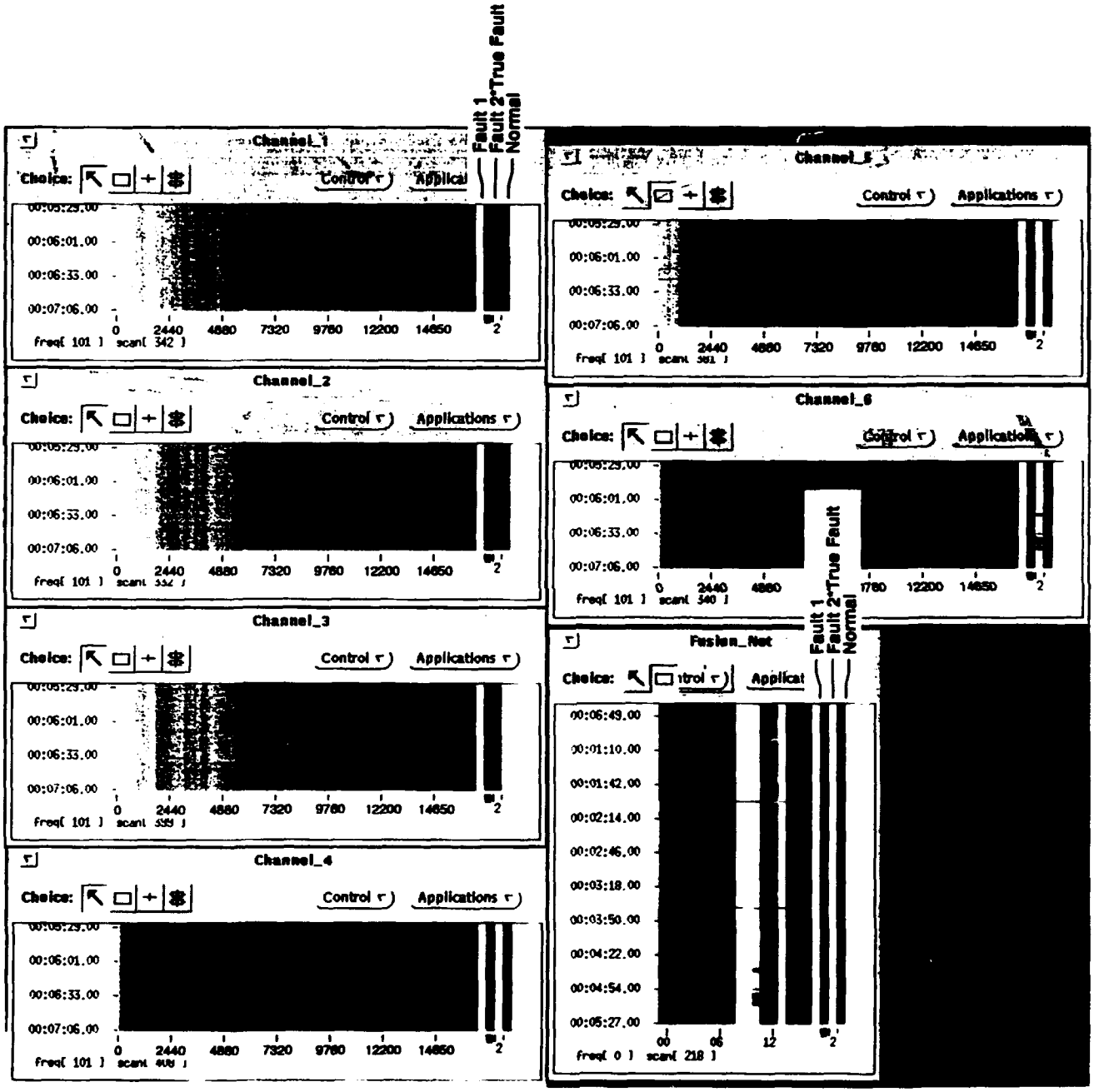


Figure 16. Condenser Pump System with Class 2 Fault

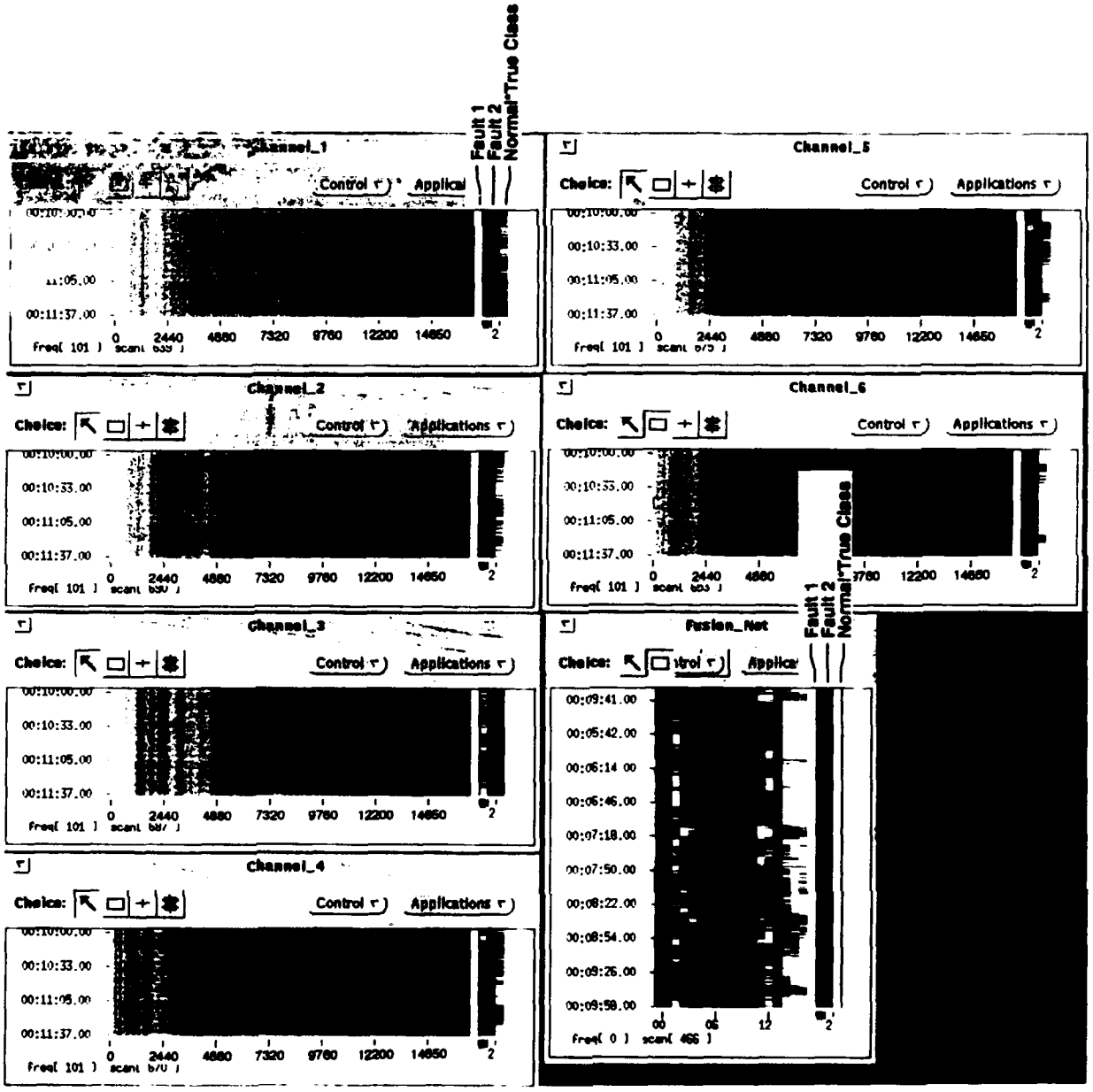


Figure 17. Condenser Pump System with Normal Data

Table 8. Condenser Pump System Performance Summary

Data	Threshold ①	P(d)	P(fa) ②	Confusion Matrix Summary
Test All Data	0	1	$<1.72\times10^{-4}$	Diagonal
Test on Separate Test Data	0	1	$<2.5\times10^{-4}$	Diagonal

① Nominal Threshold Setting = 0 (i.e., Max Fault Neuron = Normal Neuron).

② P(fa) Estimates on Per Scan Basis. When NO False Alarms in Data P(fa) = 1/(total # of normal condition scans).

FFT Feature Extractor Sufficient. For each of the three data sets, perfect performance was attained using only FFT feature extraction. The use of FFT processing with intelligently selected parameters would simplify the transition to a real-time system. However, it is possible that more complex fault detection and classification scenarios will require more advanced, and computationally expensive, feature extraction. Thus, although the high-resolution feature extractors such as the AOK TFR and Prony modeling were not needed for these data sets, they may be of use in the future, particularly in conjunction with a tachometer (sync) signal.

Necessity for Multifeature and Multichannel Processing. All three of the final systems developed benefited from multifeature (Hollins data) or multichannel processing (fire and condenser pump data). Reliance on only one feature or channel could severely degrade system performance due to data variability issues between training and testing data. The performance of the second-layer fusion net was seen to greatly improve overall performance by arbitrating between sometimes conflicting first-layer results. In addition to multichannel processing, the hierarchical approach described here can easily include additional sensor types and static information (for example flight conditions) as well.

5.0 RECOMMENDATIONS

In follow-on Phase II, ORINCON would design, develop, build, and deliver a prototype helicopter transmission fault detection and classification system to ONR for field evaluation. The system would be developed and tested using accelerometer and other sensor data from the

transmission of a CH-46 helicopter. This data would be supplied by ONR. The Phase II effort would include further research into advanced feature extraction techniques and evaluation of new classification algorithms. New classification algorithms are required to address the novelty detection problem. The general architecture of the system would be the hierarchical neural network approach used in Phase I. Figure 18 shows a generic flow diagram for such a processing system. Specific objectives to be accomplished in Phase II are listed below.

Feature Extraction. Additional feature extraction techniques that we believe have potential for better characterizing the data will be investigated. These feature extraction techniques include data adaptive STFT processing, higher-order statistical processing (bispectra), cyclostationary processing, and cross-channel coherence. Additionally, the data supplied by ONR is expected to have a tachometer signal that can be used for cycle synchronization. The use of this signal in template selection and retina generation will be examined, and the utility of extremely high-resolution time-frequency representations (such as the AOK TFR and Prony's method) will be re-examined. It is anticipated that selection of processing parameters that make use of the tachometer signal will improve high-resolution feature extractor performance.

Alternative Classifiers/Neural Nets. In Phase I, only three-layer perceptron neural networks were utilized. Although we have had outstanding results with this classifier, the multilayer perceptron (MLP) is not ideally suited to novelty detection. That is, the response of the MLP to data for which it has not been trained is not well defined[13]. We would like to investigate classifiers that behave in a more predictable manner when confronted with new data. These classifiers will indicate if the transmission is operating in an unknown, but not normal, manner, and will provide a measure of similarity between the present condition and trained fault conditions. This property will aid in the detection of untrained fault conditions and will provide an indication as to the severity of advancing degradation. The classifiers to be examined, in addition to the MLP, will be the Radial Basis Function (RBF) neural network and the Fuzzy Min-Max Neural Network (FMMNN). Because of the inherent modularity of the hierarchical neural network architecture, it will be straightforward to substitute the new classifiers for the MLP networks used in Phase I.

Quantitative System Analysis. Using the data set provided by ONR, extensive quantitative system analysis will be undertaken to evaluate statistical performance. This testing will serve two purposes. First, it will aid in the design of the system, allowing quantitative evaluation of the utility of various data channels, feature extractors, and classifiers, in the fault detection and classification solution. Testing will be performed on both the first-layer nets as well as the second-layer fusion net. After the final design of the system is established, testing will be

[13] J.A. Leonard and M.A. Kramer, "Radial Basis Function Networks for Classifying Process Faults," *IEEE Control Systems Magazine*, April 1991.

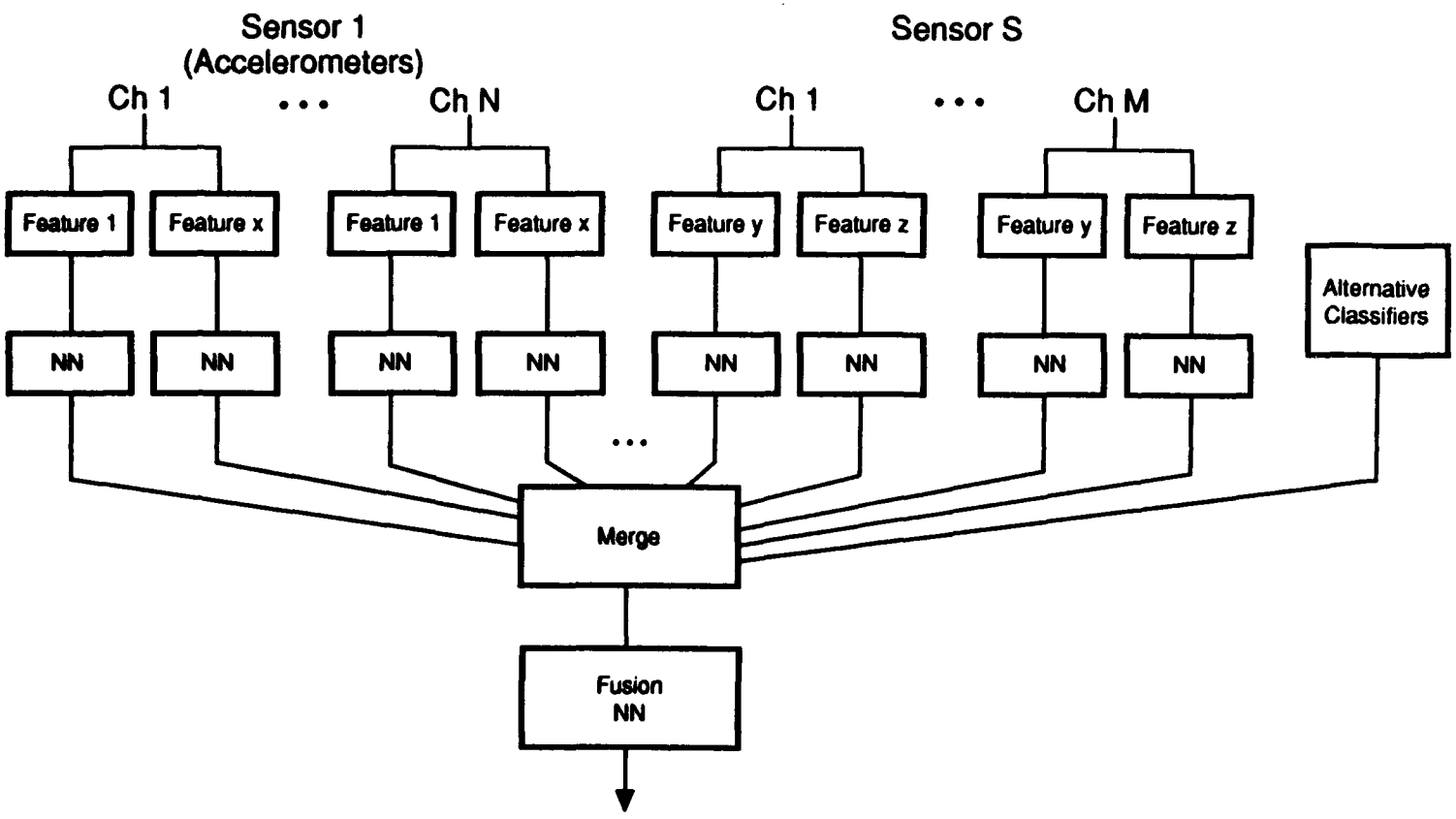


Figure 18. Phase II System for Helicopter Transmission Fault Detection and Classification

performed to establish the overall system capabilities. Performance measures to be evaluated include the probabilities of fault detection, false alarm, correct classification, and misclassification. Additionally, we would like to investigate the performance of the system as a function of fault severity or degradation advancement. This analysis is dependent upon the supply of data with varying degrees of fault severity.

System Prototype Delivery. The systems developed in Phase I were implemented using ORINCON's i860-based, real-time rapid prototyping system known as PRIISM. The Phase II research will also be completed using the PRIISM hardware and software system. At the completion of Steps 1, 2, and 3 above, the PRIISM system will be modified to accept real-time sensor data from the tachometer and transmission accelerometers of the CH-46, and the system will be delivered to ONR for field testing on the CH-46.

System Miniaturization Design. As part of the Phase II effort, a preliminary design will be completed for miniaturization of the prototype system and transition to a commercial product. One promising candidate for this design is the Signal Processor Packaging Design (SPPD) shared-memory multiprocessor developed by Rockwell. Each SPPD module can accommodate up to 16 TMS320C30 processors and eight parallel and eight serial ports (32 and 6 Mbytes, respectively). A 16-processor module can perform up to 500 MFLOPS. The package is small (68 cm^3, 75 gm) and rugged, and requires only 11 watts (5 volts) of power. The processors support the assembler, C, and Ada programming languages.

APPENDIX A

Applications of Time-Frequency and Time-Scale

Representations to Fault Detection and Classification

Presented at 26th Asilomar Conference on
Signals, Systems, & Computers, October 1992

Application of Time-Frequency and Time-Scale Analysis to Underwater Acoustic Transients

Larry Marple
Orincon Corporation
10015 Old Columbia Road
Columbia, MD 21046

Tom Brotherton , Tom Pollard,
Rick Barton and Avi Krieger
Orincon Corporation
9363 Towne Centre Dr.
San Diego CA, 92121

Abstract

Underwater acoustic transients from man-made structures and biologics are rich in structure and detail, diverse in terms of duration, and highly nonstationary,. Orincon has an ongoing program to find algorithmic analysis techniques that can extract features of these transients for purposes of high confidence classification. It has been found that no one analysis technique can adequately capture features from all the possible transient classes. Presented here are side-by-side comparisons of numerous time-frequency and time-scale representations of unclassified acoustic transients. The techniques illustrated include the Short-Time Fourier Transform (STFT) tuned to different analysis bandwidths, the Gabor -Wavelet Time-Scale Representation (TSR), the Adaptive Optimal Radially Gaussian Kernel (AOK) TFR, and model based TFRs. The strengths and weaknesses of the various analysis representations are discussed.

1. Introduction

The purpose of this paper is to present an honest side-by-side evaluation of several TFR and TSR techniques with real transient data in order to validate the performance claims found in the literature. It also provides an opportunity to see the relative performance of several TFR/TSR techniques side by side. The general conclusion is that it is hard to beat the interpretability and robustness of the the classic STFT when operating with real data.. Section 2 discusses the various processing techniques used. Section 3 provides a real data processing example. Section 4 contains conclusions.

2. The Processing

2.1 The STFT

The STFT TFR is the best understood and most robust of the various TFRs. Its linear response makes interpretation of features easy to understood. It is computationally efficient to compute with the FFT. The only handicap of the STFT is limited resolution due to the windowing relative to some fast time-varying feature. The short-time fast Fourier transform is found as follows: let x(t) represent the input data set; let $X_n(f)$ represent the FFT of the n-th segment of data. That is,

$$X_n(f) = \frac{1}{N}\sum_{t=0}^{N-1} w(t)\, x(nT+t)\, e^{-j\frac{2\pi ft}{N}} \tag{1}$$

where typically T = N/2 (i.e., "50% overlap") and N is a power of 2. w(t) is a window function, such as the Hamming window. Multiple STFT scans are appended together to form the TFR.

2.1 The Short Time Autoregressive (AR) Model

The short-time autoregressive transform is similar to the STFT, except the modified covariance algorithm [1] is substituted for the windowed Fourier transform at each time increment. For each segment normally evaluated by the STFT, the AR power spectral density function replaces the STFT computation:

$$X_n(f) = \frac{T\rho_w}{\left|\sum_{m=0}^{M} a[m]e^{-j2\pi fmT}\right|^2} \tag{2}$$

in which the AR coefficients a[m] are computed by means of a least squares algorithm from the data samples. The AR technique can provide a higher resolution over the STFT without the complication of the quadratic terms found in the quadratic TFRs or the need to generate a high time-resolution scaling analysis wavelet of the TSRs.

2.1 The Singular Value Decomposition (SVD)

The SVD approach uses the eigenvectors associated with the maximal eigenvalues of the data covariance matrix. If x(t) is assumed to be the sum of a signal process and noise

$$x(t) = s(t) + e(t), \tag{3}$$

then the covariance matrix R is given by

$$R = R_s + R_e \tag{4}$$

where R_s and R_e are the signal and noise covariance matrices. Assuming that the background noise is white, then the noise covariance matrix $R_e=\sigma^2 I$. Using the SVD, R can be written as

$$R = \sum_{k=1}^{n}(\lambda_k - \sigma^2)V_k V_k^T + \sigma^2 I \quad . \tag{5}$$

λ_k are the eigenvalues and V_k are the associated eigenvectors and there are assumed to be n eigenvalues / eigenvalues associated with the signal. Equation(5) is the key to the subspace representation approach that is exploited by the MUSIC algorithm. Denote a matrix T_S as

$$T_s = \sum_{k=1}^{m} V_k V_k^T \quad . \tag{6}$$

The result of using T_S on any vector in R^n characterizes the signal subspace and therefore the signal itself. For display purposes, we use the Fourier transform of V_k. Here we assume that m=1. This representation is well suited for single sine waves. We have mainly used this representation for detection; the SVD spectra for any signal is significantly different then that of white noise.

2.1 The Bispectrum

The bispectrum of a stationary random process conveys information about the third-order cumulant structure of the process. The bispectrum is the two-dimensional Fourier transform of the two-dimensional third-order cumulant sequence. For a zero-mean random process the bispectrum $B(\omega_1,\omega_2)$ is defined by

$$B(\omega_1,\omega_2) = \sum_{n_1=-\infty}^{\infty}\sum_{n_2=-\infty}^{\infty} C_3(n_1,n_2)e^{-i(\omega_1 n_1+\omega_2 n_2)} \quad , \tag{7}$$

where $C_3(n_1,n_2)$ is the third-order cumulant sequence.

The bispectrum has many useful properties, but perhaps the most important is that the bispectrum of a stationary Gaussian random process is identically zero. Since background noise is generally a composite signal that approximately Gaussian, its bispectrum is near zero. On the other hand, the nonlinearities present in many of the mechanisms that generate real-world signals of interest often induce significant non-Gaussian structure, and many such signals exhibit non-zero bispectra. In such cases, detection and classification performance can often be enhanced by operating on bispectral statistics rather than second-order statistics. In effect, the bispectral statistics exhibit an increased signal-to-noise ratio.

Here the bispectrum estimates are computed for M = 520 points uniformly distributed over the principal domain of the bispectrum. The estimate is computed by forming a weighted average of a number of *generalized periodograms*, which are given by

$$I(j,k) = \frac{1}{N}X(j)X(k)X^*(j+k) \quad , \tag{8}$$

where j and k are integers and "*" indicates complex conjugation. To precisely define the averaging operation, we make use of a frequency-domain weighting function W, which can be any nonnegative function of two variables that satisfies the constraints

(i) $W(\omega_1,\omega_2) = W(-\omega_1,-\omega_2)$, and

(ii) $\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} W(\omega_1,\omega_2)d\omega_1 d\omega_2 = 1.$ (9)

Here a quadratic weighting function was chosen. Given any such weighting function, we define the scaled version W_N by

$$W_N = N^{2-2c}W\left(N^{1-c}\omega_1, N^{1-c}\omega_2\right), \tag{10}$$

where c is a parameter in the range $0 < c < 1$. Adjusting the value of c alters the bias-to-variance ratio of the bispectrum estimates. We also make use of a special frequency-domain window function Φ, defined by

$$\Phi(i,j) = \begin{cases} 0 & \text{if } i \equiv 0(\text{mod } N), \\ & j \equiv 0(\text{mod } N), \text{or} \\ & i+j \equiv 0(\text{mod } N) \\ 1 & \text{otherwise} \end{cases} \quad .$$

This function is used to suppress the undesirable influence of the generalized periodograms at a subset of points in the frequency domain. The bispectrum estimate is

$$\hat{B}(\omega_1,\omega_2) = \left(\frac{2\pi}{N}\right)^2 \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} W_N\left(\frac{2\pi i}{N}-\omega_1, \frac{2\pi j}{N}-\omega_2\right)\cdot\Phi(i,j)\cdot I(i,j) \tag{11}$$

The entire collection of M bispectrum estimates forms an M-dimensional vector. The M points in each vector are arranged in order of increasing value for each of the frequency components, with the first component ω_1 varying most rapidly.

2.2 The AOK TFR

An instantaneous time / frequency representation (TFR) gives a high resolution characterization of the data in time as well as FFT resolutions in frequency for signals of interest. The particular TFR that we use here is the adaptive optimal radially-Gaussian kernel TFR developed by Baraniuk and Jones [2][3]. The TFR uses a radially-Gaussian *signal-dependent* kernel that changes shape to optimally smooth the distribution.

The optimal kernel, Φ, for a signal is defined as the solution to the following optimization problem:

$$\max_{\Phi} \int_0^{2\pi}\int_0^{\infty}\left|A(r,\psi)\,\Phi(r,\psi)\right|^2 r\,dr\,d\psi \tag{12}$$

subject to

$$\Phi(r,\psi) = e^{-\frac{r^2}{2\sigma^2(\psi)}} \tag{13}$$

$$\frac{1}{2\pi}\int_0^{2\pi}\int_0^{\infty}\left|\Phi(r,\psi)\right|^2 r\,dr\,d\psi \le \alpha, \quad \alpha \ge 0 \tag{14}$$

$A(r,\Psi)$ is the ambiguity function (AF) of the signal in polar coordinates. Once the optimal kernel is computed, the TFR is given by

$$P(t,\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A(\theta,\tau)\,\Phi(\theta,\tau)\,e^{-j\theta t - j\tau\omega}\,d\theta\,d\tau \tag{15}$$

The representation is good for characterizing short duration and nonstationary events. The AOK TFR is computationally expensive. As with the STFT feature extractor, a time sequence of the AOK TFRs form the input retina.

2.3 The Prony Model Method

Prony's model method assumes the signals of interest are modeled by a sum of damped sinusoids. The model is well suited for characterizing impulsive type of events [5]. The resulting model gives a variety of parameters that may be exploited for characterizing transient waveforms. The Prony Model is of the form :

$$x[n] = \sum_{k=1}^{p} A_k \exp\left[(\alpha_k + j2\pi f_k)(n-1)T + j\theta_k\right] \tag{16}$$

where x[n] is the observed time series data, p is the model order, A_k is the amplitude of the k-th coefficient, α_k is the corresponding damping term, f_k is the center frequency, T is the sample interval, and θ_k is the initial phase. The parameters of the model can be estimated using least squares techniques [1]. Several different noise discrimination techniques have been developed for use with the Prony model method. In the processing presented here a time sequence of spectral estimates similar to those defined for the AR model in equation (2) are computed using the Prony model parameters.

2.4 Wavelet Processing

The wavelet transform (WT) is a time domain representation of a signal in terms of dilated and shifted versions of suitable analyzing wavelets. The wavelet transform of a function f(x) represents a decomposition of the function in terms of dilated and shifted versions of an analyzing wavelet function $\psi(x)$. The transform is linear, energy preserving, and invertible so for every function f(x), there is a unique, continuous 2-D transform Wf(s,u). The wavelet transform can also be regarded as a time-frequency representation of the signal, in which the parameter $s > 0$ corresponds to frequency and the parameter $u \in R$ corresponds to temporal shift. The wavelet transform has many interesting properties that make it particularly well suited as a signal representation [6].

The wavelet transform can be viewed as a time-frequency map of the signal in which the frequency information is generated by a bank of proportional-bandwidth filters. Because of the inverse relationship between bandwidth and temporal support, the wavelet time-frequency representation automatically provides greater temporal resolution for high-frequency signal components. This is often very useful when analyzing transient or highly nonstationary phenomena.

The wavelet features presented herein were generated using an analytic Gabor wavelet. The wavelet is defined in the frequency domain by the following equation

$$H(\omega) = \begin{cases} \frac{1}{2i}\left(e^{-\frac{\sigma^2(\omega-\omega_0)^2}{2}} - e^{-\frac{\sigma^2(\omega+\omega_0)^2}{2}}\right) & \text{if } \omega \ge 0 \\ 0 & \text{if } \omega < 0 \end{cases} \tag{17}$$

where $\omega_0 = 2\pi$ and $\sigma = 6.1182$. This choice of parameters results in a time-frequency map in which the bandwidth of each frequency bin is approximately 1/16th of an octave.

3. Real Data Processing Results

Two data sets are considered; the dropping of a wrench. and a whale call. Both events are complicated time-varying signals that are difficult to analysis. The sample rate was 32000 samples per second.

The STFT TFR uses 64-points per scan with 56 points of overlap between scans. The data was windowed using a Hamming window and zero padded to generate a 512 point transform.

The AR TFR uses a 64-points analysis window with 56 points of overlap and order 20 AR/linear prediction filter. The AR function is evaluated at 512 points in the frequency domain using an FFT.

The Prony model uses 64-point analysis window with a 56 point overlap and an order 16 for the wrench and order 20 for the whale. The Prony model is evaluated at 512 points in the frequency domain with an FFT transform of the model coefficients.

The AOK TFR uses 512 points in the analysis window for each scan. The data overlap between scans is 504 points (i.e. 8 points are skipped between scans). The kernel volume constraint was set to $\alpha = 2.25$.

The SVD algorithm uses a 512 point FFT on the largest eigenvalue eigenvector. The algorithm used an 8x8

matrix which was averaged 16 time each pass through. 32 new samples were brought into each new pass.

The bispectrum uses an analysis window of 128 points and c=0.5 was chosen. These parameters give an averaging region which uses approximately 4 generalized periodograms.

The wavelet transform uses differing numbers of samples for each frequency bin output. Here 256 bins were generated beginning with a center frequency of 400 Hz. and increasing geometrically to the Nyquist frequency. The time shift variable was incremented 20 samples for each step.

Figures 1 and 2 show the various TFRs / TSRs. The data values are encoded using a grayscale; white pixels are relatively large values and black pixels have low values.

4. Conclusions

For the most part it is hard to beat the STFT for feature interpretation. However there are unique features that can be determined in all of the different representations.

References

[1] S.L.Marple, Digital Spectral Analysis with Applications, Prentice-Hall, 1987

[2] R.G.Baraniuk and D.L.Jones, A Radially Gaussian, Signal-Dependent Time-Frequency Representation, *IEEE ICASSP '91*, May 1991.

[3] R.G.Baraniuk, D.L.Jones, T.W.Brotherton, and S.L.Marple, Applications of Adaptive Time - Frequency Representations to Underwater Acoustic Signal Processing, *25th Asilomar Conference on Signals, Systems & Computers*, Pacific Grove CA, Nov. 1991

[4] Hlawatsch and G.F. Boudreaus-Bartels,"Linear and Quadratic Time-Frequency Signal Representations", IEEE Signal Processing Magazine, vol. 9, pp. 21-67, April 1992.

[5] S.L.Marple and T.W.Brotherton, Detection and Classification of Short Duration Underwater Acoustic Signals by Prony's Method, *IEEE ICASSP '91*, May 1991.

[6] Rioul, O. and Vetterli, M. (1991), "Wavelets and signal processing," IEEE Signal Processing Mag., Oct. 1991.

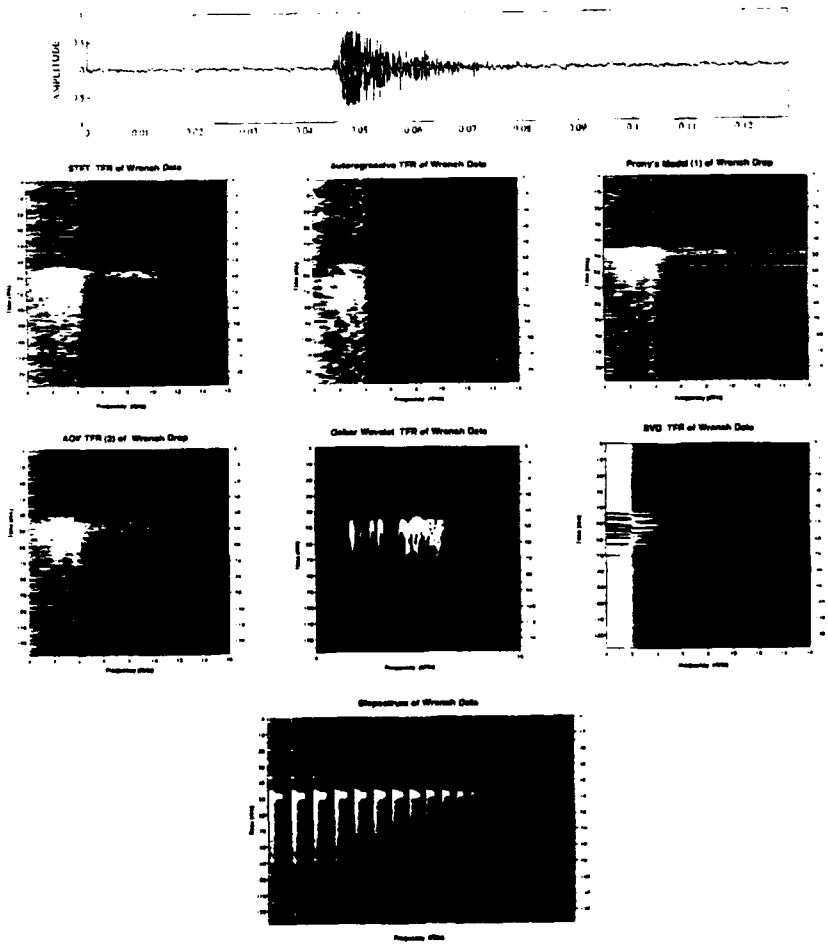


Figure 1. Various TFRs for Wrench Drop Data

COPY AVAILABLE TO DTIC DOES NOT PERMIT FULLY LEGIBLE REPRODUCTION

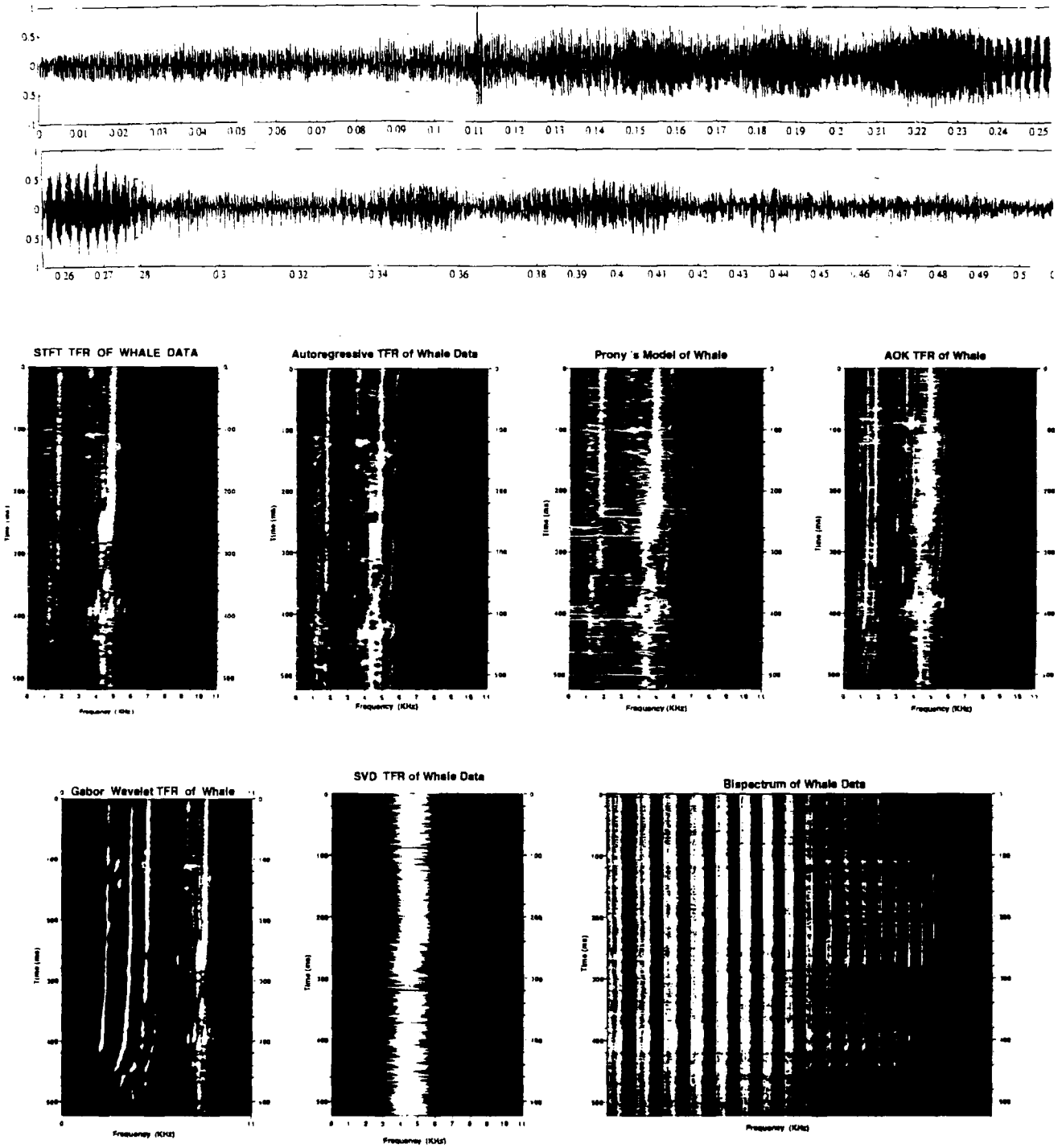


Figure 2. Various TFRs for Whale Call Data

COPY AVAILABLE TO DTIC DOES NOT PERMIT FULLY LEGIBLE REPRODUCTION