Abstract—This paper has three contributions. First, we develop a low-cost test-bed for simulating bearing faults in a motor. In Aerospace applications, it is important that motor fault signatures are identified before a failure occurs. It is known that 40% of mechanical failures occur due to bearing faults. Bearing faults can be identified from the motor vibration signatures. Second, we develop a wireless sensor module for collection of vibration data from the test-bed. Wireless sensors have been used because of their advantages over wired sensors in remote sensing. Finally, we use a novel two-stage neural network to classify various bearing faults. The first stage neural network estimates the principal components using the Generalized Hebbian Algorithm (GHA). Principal Component Analysis is used to reduce the dimensionality of the data and to extract the fault features. The second stage neural network uses a supervised learning vector quantization network (SLVQ) utilizing a self organizing map approach. This stage is used to classify various fault modes. Neural networks have been used because of their flexibility in terms of online adaptive reformulation. At the end, we discuss the performance of the proposed classification method.

1. INTRODUCTION

Failure avoidance is one of the main approaches for ensuring the quality and performance of a system. There are two main types of failure avoidance in terms of maintenance, namely preventive and corrective. In preventive maintenance, all actions are taken to keep equipments in good operating condition. It should be able to indicate when a failure may occur so that actions can be taken to avoid failures. In corrective maintenance, a repair is performed after a failure has occurred. Condition-based maintenance (CBM) is an approach of preventive maintenance. The process of CBM involves monitoring the system, predicting failures and making repairs before these failures occur. A system can contain many fault modes and a decision has to be taken on the type of repair necessary for eliminating any future faults [7] [34] [43].

Monitoring of the system is done using a range of sensors which can either be wired or wireless. Wireless sensors are generally used to enable remote monitoring. Wireless Sensor Networks (WSN) provide an intelligent platform to gather and analyze data without human intervention. Typically, a sensor network consists of autonomous wireless sensing nodes that are organized to form a network. Each node is equipped with sensors, embedded processing unit, short-range radio communication module, and power supply, which is typically 9-volt battery. With recent innovations in MEMS sensor technology, WSN hold significant promise in many application domains [42].

In this paper, we concentrate on CBM for induction motors. In Aerospace applications, it is important that motor fault signatures are identified before a failure occurs. Some of the common problems in induction motors are: bearing, stator winding, and rotor bar failures. Since the bearings carry the weight of the rotor, its fault diagnosis becomes very important. A multi-layered feed forward Neural Network (NN) trained with Error Back propagation technique and an unsupervised Adaptive Resonance Theory-2 (ART2) based NN has been used for detection and diagnosis of localized faults in ball bearings in [41]. In [25] Auto-Regression Model with NN has been used for fault detection.
diagnosis. In this paper, we have made use of a two stage Neural Network through which online adaptive reformulation is possible. This method eliminates the need for the entire set of data to be present for classification as opposed to back propagation method and hence, saves on the processing time if the volume of data is large. Principal Component Analysis for feature extraction used in the first stage of NN is advantageous compared to statistical methods as it avoids batch processing. There is no compromise on resolution as in the case of AR modeling. The proposed algorithm works well both with linear and non-linear systems in comparison to Radial Basis Function and back propagation methods, which work best when the system is linear.

In this paper, we concentrate on motor bearing fault diagnosis and prognosis. It has been shown that 40% of machine failures occur due to bearing problems. Since the mechanical faults are the faults pertaining to the bearings of the machine, their maintenance is very important [36]. The causes for the bearing faults may be:

**Mechanical Damage**: Improper handling of the bearings results in dents and nicks, causing displacement of the metal particles which can introduce secondary effects in the motor.

**Damage due to wear**: Wear and tear in the bearings may result in a gradual deterioration of the motor. These conditions may become prominent for the bearing failure in the long run.

**Corrosion Damage**: This damage comes into picture when the motor is operated in a moist atmosphere. The moisture in the air causes surface oxidation and rusting, paving the way for abrasion and crack initiation.

**Crack Damage**: This occurs when the motor is subjected to large stresses through overloading or cyclic loading. Cracks may also be produced because of manufacturing defects and improper heat treatment and grinding.

**Electric Arc Damage**: Bearings can be damaged when the grounding of the equipment is not done properly. An arc is produced between non-contacting elements and its runways when current passes through it.

**Damage due to lubricants**: It is caused due to improper lubrication system of the bearings which is present between the rolling elements and the raceways. More details can be obtained from [36].

In this work, we design a test-bed for simulating bearing faults in a motor. We design a custom wireless sensor module for gathering vibration data from the motor. We use a two stage neural network design to classify and diagnose different fault modes from the collected vibration data. The first stage neural network uses the Generalized Hebbian Algorithm (GHA) to estimate the principal components of various fault modes [12]. Principal Component Analysis is used for two reasons: a) to reduce the dimensionality of the data and b) to extract the features. Neural Networks is preferred for PCA estimation rather than statistical methods because of its flexibility in terms of online adaptive reformulation.

The second stage neural network uses a supervised learning vector quantization (SLVQ) network utilizing a self organizing map approach. This stage is used to classify various fault modes. The output of the first stage neural network is fed as an input to the second stage neural network.

The paper is organized as follows: Section II describes the construction of the test-bed for simulating various bearing faults. It also describes the design of the wireless sensor module used for gathering vibration data from the test-bed. Section III describes the two stage neural network architecture for fault diagnosis. Section IV shows the results of this work describing the performance analysis. Section V concludes our work.

## 2. The Test-Bed

This section describes the construction of the test-bed used for simulating mechanical bearing faults in a motor. It also describes the design of the wireless sensor module used for collecting vibration data from the test-bed.

### A. Test-bed for Bearing Fault Simulation

For a particular bearing geometry, the rolling elements in the bearing produce vibration spectra which have unique frequency components. These frequency components along with their magnitude help in determining the condition of the bearing. The test-bed for the mechanical fault consists of a motor to which a flywheel is attached (Figure 1).

This is enclosed in a metallic cage for the purpose of safety (Figure 2) when the loads are attached to it. The most important element of this test-bed is the flywheel which has holes drilled in it (Figure 3b). The weights applied to these holes produce imbalance in the flywheel, thereby in the motor. The flywheel actually simulates the corrupted bearings of the motor. The holes are drilled in three concentric rings: inner, middle and outer as shown in Figure 3a.

The severity of the fault is determined by the location of the weight- the highest being when the weight is applied on the outermost ring and vice versa. The disc, therefore acts as the load to the motor, with the screws (weights) simulating the mechanical fault at the bearings of the motor.

The increase in the addition of the weight increases the frictional loss which in turn would increase the total loss in the motor and hence reduces the efficiency of the machine. Another reason for the imbalance in the motor is also a shift in the center of gravity of the flywheel when weights are added to it.

The motor rotates in order to reduce the effect of this shift so that the entire weight acts through the center of the body. The test-bed described above gives an economical method to emulate the bearing fault behavior.

### B. Wireless Sensors

Wireless sensors are used for measuring the vibration data from the test-bed. A custom PCB, designed at
Automation and Robotics Research Institute, has been used for the wireless sensing. The remote sensing board contains a Microchip PIC 18F4550 microprocessor running at 20 MHz connected to a small, low power, 3-Axis ±3g iMEMS Accelerometer (ADXL330) and a Jennic wireless microcontroller module. This module transmits in the 2.4 GHz spectrum using the 802.15.4 wireless protocol. In this setup, the PIC microcontroller is dedicated entirely to collecting the samples using its onboard analog to digital peripheral and then transmitting them through its UART peripheral to the Jennic module for wireless transmission to the computer. The Jennic is responsible for receiving the incoming serial data and transmitting it as quickly as possible. Previously attempts were made to use only the Jennic which has its own analog to digital peripheral, but the module by itself was too slow to handle the data collection and transmission functions together.

The receiving end consists of another Jennic module that receives the samples and outputs a digital representation on the UART. The UART of the Jennic module is connected to the computer's serial port via a TTL-RS232 converter. The firmware for this project consists of the programs running on both the PIC and the Jennic modules. The Jennic firmware was created using the freely provided Jennic SDK. The firmware is based on a Jennic provided application note that provides interrupt-driven transmission of serial data over the wireless radio. On the receiving end, the firmware on the Jennic formats the sample data in a manner to make it easier to parse and use by the computer application. The PIC firmware was developed using the CCS compiler for mid-range PIC microcontrollers. This firmware was optimized to collect 8-bit samples at a rate commensurate with the bandwidth of the RF link. As the samples are collected at the desired sample rate, they are stored in a FIFO buffer and sent to the Jennic module as the UART becomes available.

This wireless board also has many other capabilities not utilized in this project. Originally, it was designed and used for autonomous aerial vehicle control. In addition to collecting analog data, it is also capable of interfacing to sensors that use simple general purpose IO or synchronous serial busses including I2C. It is also capable of driving its IO pins to logic levels and generating PWM signals for possible control applications. Figure 4 shows the wireless modules used for the experiment. Figure 5 shows the arrangement of the sensor for recording the three dimensional vibration data from the motor. This arrangement is chosen to maintain uniformity in x, y and z values obtained each time.
3. THE TWO-STAGE NEURAL NETWORK CLASSIFIER

The analysis of the data obtained is done using a novel two stage neural network, where, the first stage is used for estimation of Principal Components and the second stage for classification [5].

A. First Stage: PCA estimation

The dimensionality of the large amount of data recorded is reduced using Principal Component Analysis. This reduction is done by retaining only those values in the data set which have a significant value of variance. The main advantage of this method is that the information content is retained despite a reduction in the dimensionality. PCA is used for two reasons: a) to reduce the dimensionality of the data and b) to extract the features. Hebbian Learning Rule Neural Network is used for PCA estimation. Information on PCA estimation using GHA can be found in [12]. It is used because it extracts m actual principal eigenvectors as those obtained from the conventional method. The conventional method consists of finding the mean of the data set, then subtracting the mean from each of the data values, calculating the covariance matrix and then finding the eigenvalues and eigenvectors of the covariance matrix. The calculation of eigenvectors and eigenvalues of the covariance matrix helps in extraction of those lines which characterize the data. The eigenvectors calculated are then arranged in a descending order, after which the components with lesser significance can be ignored. The chosen eigenvectors are then used to form the feature vector in the columns. Further information can be found in [24]. Hence, PCA compresses the data \( X \in \mathbb{R}^{n \times d} \) and gives a lower dimension of \( y = WX \) where \( y = \mathbb{R}^{n \times d} \). Here, \( W \) matrix represents the eigenvectors chosen.

Neural Networks is preferred for PCA estimation rather than statistical methods because of its flexibility in terms of online adaptive reformulation [12]. The conventional method involves calculation of the covariance matrix and then application of the diagonalization procedure for extracting the eigenvalues and corresponding eigenvectors. As the size of the data increases, matrix manipulation and computation becomes cumbersome and inefficient due to round off errors. Hence, this poses a limitation on the statistical based methods.

![Figure 4. (a) Jennic Wireless Module, (b) PIC 18F4550 Module](image)

![Figure 5. The Conceptual Diagram showing the alignment of the sensor with respect to the motor](image)

Let us take an example in order to understand the difference between the 2 approaches: The dimension of the vibration data obtained is: \( [3, 16000]^{T} \) (3 corresponds to the x, y, and z axis vibration readings and 16000 corresponds to the data points) and suppose the total number of data sets is \( M = 100 \) for a particular fault. The data points obtained are converted to 1D vector given by \( \tau \), corresponding to the data point \( i \). The dimension of \( \tau \) is \( \{N,1\} \) where \( N = 3 \times 16000 = 48000 \). The mean of the data set is

\[
\psi_i = \frac{1}{M} \sum_{i=1}^{M} \tau_i = \frac{1}{M} [\tau_1 + \tau_2 + \ldots + \tau_M]
\]

And subtracting the mean from the original data gives the difference value as

\[
\phi_i = \tau_i - \psi
\]

Where \( i = 1, 2, 3, \ldots, M \).

\[
A = [\phi_1, \phi_2, \ldots, \phi_M]
\]

Where the dimension of \( A \) is \([48000,100]\). The covariance matrix \( C \) is obtained as

\[
C = \frac{1}{M} \sum_{i=1}^{M} \phi_i \phi_i^T = AA^T
\]

The eigenvectors of \( C \) are computed to form the transformation matrix \( W \). It can be observed that the dimension of the matrix \( C \) is \( \{N,N\} \) which makes it cumbersome to compute the \( N \) eigenvectors. As the number of eigenvectors giving information about the data is equal to the number of training data sets over which the information is obtained; therefore, only \( M \) eigenvectors are used to form the \( W \) matrix. In order to compute the \( M \) eigenvectors the following method is used, where a new matrix is given as

\[
L = A^T A
\]

Now, the dimension of \( L \) is \( \{M,M\} \). The complete formulation of the principal components is given in [14]. Through this way, only \( M \) eigenvectors are computed instead of computing \( N \) eigenvectors. One of the main disadvantages of using the statistical based method to determine the principal components is that it is based on batch processing, where all the information needs to be known \textit{a priori}. Even if a new data set comes into the
system architecture then in order to update the principal components, the entire procedure mentioned from (1)-(4)-needs to be carried out. In order to avoid this problem, Neural Networks are used in stage 1 to estimate the principal components of the data set, instead of the original matrix algebra method, based on batch computation. The advantage of using NN instead of batch matrix computation to calculate the principal component is that it gives the flexibility of online adaptive reformulation as well as recursive updating of the principal components. Using NN for PC estimation does not require having all the information at once i.e. batch processing is avoided. As the data sets are assimilated one by one, the PC are updated sequentially. If a new data set appears to the system architecture, its size is reduced by the PC estimated earlier. A bank of NN is used for each fault condition, which makes the system robust.

The neural network deployed in NN banks is based on Sangers rule [12] and uses Generalized Hebbian Algorithm (GHA) to estimate the principal components. The main advantage of using the Sangers rule is that the network naturally orders the principal components by magnitude. As mentioned earlier, the estimation of the principal components is basically the determination of the transformation matrix \( W \). The determination of the transformation matrix \( W \) using the Sangers rule, in scalar form is given as

\[
\mathbf{w}_j(k+1) = \mathbf{w}_j(k) + \mu(k) \mathbf{y}_j(k) \left[ x_j(k) - \sum_{h=1}^{i} \mathbf{w}_h(k) \mathbf{y}_h(k) \right]
\]

(6)

where \( i = 1,2,\ldots,m \) and \( j = 1,2,\ldots,N \). \( m \) is the number of principal components to be estimated and \( N \) is the number of input vectors. The weight update equation in (6) is derived by minimizing the performance index given as

\[
J_1(X) = \frac{1}{2} (X - \hat{X})^T (X - \hat{X})
\]

(7)

\[
= \frac{1}{2} (X - W^T y)^T (X - W^T y)
\]

\[
= \frac{1}{2} (X - W^T W X)^T (X - W^T W X)
\]

with respect to \( W \), using the steepest descent gradient method [22]. \( X \) is the input and \( y \) is the output (as in \( y = WX \)). Further variations of the update rules are given in [23, 24]. The reason of using Sangers rule based on GHA is that it extracts the \( m \) actual principal eigenvectors as those obtained from original matrix algebra method.

B. Second Stage Neural Network

The second stage is used to classify the input data where the output of the first stage becomes the input to the second stage. The classification step only requires distinguishing between different classes of faults by modeling the class boundaries. This type of learning is called as discriminative learning [14] and it does not require estimating the class feature densities. To achieve discriminative learning a supervised learning vector quantization (LVQ) network is used. LVQ is a hybrid network and uses a self organizing map approach. It is based on winner-take-all policy and uses training vector to distinguish the different categories of the input. Since LVQ network tends to have shorter training time than the backpropogation or RBF, and processing time being an important constraint in the approach mentioned, LVQ network proves to be advantageous over backpropogation and RBF. In reference [26], the performance of these networks has been compared.

The supervised LVQ network is trained using standard Kohonen learning rule. More details on this rule are given in [14]. In LVQ network (Figure 6) each neuron in the first layer is assigned to a class and then each class is assigned to one neuron in the second layer.

4. IMPLEMENTATION AND PERFORMANCE ANALYSIS

A. Implementation

The implementation involves recording the vibration data from the motor over a period of five days. The vibration data is measured for conditions of no weight as well as for weights of 5g, 10g, 15g, 20g applied on the inner, middle and outer rings of the flywheel respectively. Each of the experiment (for a particular weight and its position on the flywheel) is performed five times. The data recorded is then used for PCA estimation and prediction of the motor condition for fault diagnosis. Figure 7 shows the classification of the input data after passing through the artificial neural network for different weights, where the \( x, y \) and \( z \) axes represent the \( x, y \) and \( z \) principal component vector of the vibration data respectively. For this experiment, we have only considered the highest principal component for each fault case. A classifier maps a list of measured features into a classification state. The second stage NN Classifier actually implements nonlinear decision boundaries which uses “\( K \)-nearest neighbor” discriminant
function analysis to benchmark the neural network’s performance. The nearest neighbor classifier can be explained as follows: If it is required to classify an unknown vector \( x \), where a number of classes are represented by sample points in a feature space vector, then this is done by finding the point closest to \( x \) and assigning the closest point’s class to it. In \( K \)-nearest neighbor, nearest \( K \) points to \( x \) are found, and \( x \) gets assigned the class which is represented by the largest number of neighboring points. This classification of data paves the way for formulation of confusion matrix (with the help of testing data) and finally the Operating Characteristics.

**B. Performance Analysis**

We make use of the confusion matrix to evaluate the algorithm. We also make use of health index for evaluating the performance of the machine when weights are added to various regions of the flywheel.

1. Confusion Matrix

Confusion matrix is a visualization tool used for performance evaluation. Figure 8 shows one such confusion matrix. Details on this can be obtained from [5] [43] [44]. It gives a degree of correlation between various parameters such as features, classes, etc. It is a square matrix where the entries represent the degree of correlation between it’s \( i^{th} \) and \( j^{th} \) element. The number of rows and columns is equivalent to the number of classes. The columns represent the system’s classification whereas the rows represent the true classification. For a perfect system, only the diagonal elements will be present in the confusion matrix, whereas, the presence of off diagonal elements indicates misclassification. The sum of all the elements is equal to the number of testing data. The entries \( c_{ij} \) give the data which are actually of class \( i \) but are classified as class \( j \) by the Neural Network System. Thus, it allows the user to understand the manner in which the classes have been confused with each other.

True Positive (TP) gives the number of correct predictions; False Positive (FP) gives the number of incorrect predictions; False Negative (FN) gives the number of misses and True Negative (TN) is the number of correct rejections. Out of all the recorded data, some data sets were chosen for testing the performance of the two stage neural network. This method of formulating the Confusion Matrix is called the Generalization Test. After training, the testing data set of known categories is passed through artificial neural network classification system in order to find the number of correct and incorrect classifications for each class for getting the TP and FP rates. Table 1 shows the confusion matrix for the testing data from which the true positive rate (hits) and false positive rate (wrong predictions) can be determined to get the Operation Characteristics (OC) as shown in Figure 9. The OC is a plot of the classifier’s TP rate (sensitivity) against the FP rate (specificity). High sensitivity means that the classifier identifies most of the positive samples and its performance is good, whereas high specificity means that the classifier identifies most of the negative samples and its performance is poor. The curve always goes through two points (0,0) and (1,1). (0,0) is the point at which no positives are found by the classifier, whereas (1,1) gives the point where the classifier finds everything as positive. At (0,0), the classifier identifies all the negative cases correctly but the positive cases are identified incorrectly, whereas at (1,1), the classifier identifies all the positive cases correctly but the negative cases get identified incorrectly. The graph shows...
the OC for each of the five days during which the experiment was performed. We can see from Figure 9 that the classification performance deteriorates from 93% to 76% by experiments from day 1 to day 5 respectively. This means that the condition of the motor deteriorates each day.

Figure 9. Classification Performance

2. Health Index

Health index is also plotted to evaluate the performance of the machine when the weights are added to the various regions of the metallic disc simulating the faulty bearing. The health index as the name indicates tells the state of the machine under a particular loading condition. Further information can be obtained from [7]. It checks the condition of the motor under no load and then uses a fault dictionary to facilitate the use of feature recognition techniques to diagnose the faults. Signal behavior characteristics can be used to study the condition of the motor under no-load and loaded condition by which we can get the health index to get the deviation from the normal condition.

Table 1: Confusion Matrix for Testing Data

<table>
<thead>
<tr>
<th></th>
<th>False 5</th>
<th>False 15g</th>
<th>False 25g</th>
<th>False 35g</th>
<th>False 45g</th>
<th>Inner 5g</th>
<th>Inner 15g</th>
<th>Inner 25g</th>
<th>Middle 5g</th>
<th>Middle 15g</th>
<th>Middle 25g</th>
<th>Outer 5g</th>
<th>Outer 15g</th>
<th>Outer 25g</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predict</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classify</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>False 5</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>False 15g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>False 25g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>False 35g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>False 45g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Inner 5g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Inner 15g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Inner 25g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Middle 5g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Middle 15g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Middle 25g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Outer 5g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Outer 15g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Outer 25g</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

The health index is calculated using,

\[
HV_{\text{normal}} = \frac{1}{N} \left( \sum_{j=1}^{N} \left( \frac{\text{normal}_j - Y_n \times \text{factor}}{Y_n \times 0.1} \right)^2 \right)^{\frac{1}{2}} \]

(8)

Where,

\[
j = 1, 2, ..., N
\]

\[
Y_n = \text{mean}
\]

\[
\text{normal}_j = i\text{th sampled response}
\]

Thus, \( HV_{\text{normal}} \) is the no load value which acts as the reference for the healthy system.

For the loaded system, the equation would be given by,

\[
HV_{\text{load}} = \frac{1}{N} \left( \sum_{j=1}^{N} \left( \frac{\text{load}_j - Y_n \times \text{factor}}{Y_n \times 0.1} \right)^2 \right)^{\frac{1}{2}} \]

(9)

The health index of the system can therefore be calculated as

\[
HI = \begin{cases} 
\frac{HV_{\text{normal}}}{HV_{\text{load}}} & \text{if} \quad HV_{\text{normal}} < HV_{\text{load}} \\
1 & \text{if} \quad HV_{\text{normal}} > HV_{\text{load}} 
\end{cases}
\]

(10)

(11)

It is clear from above that the health index would range between Zero (poor) and 1 (good), giving an indication of the health of the system. For the experiment performed in the laboratory, it was found that the health index was larger for smaller values of weights applied on the inner ring compared to the larger value of weights which were applied on the outer ring. This also implies that the vibrations produced are lesser when the weights are applied on the inner ring as opposed to those applied on the middle and outer rings respectively. This would help condition monitoring systems to be implemented, where maintenance action can be brought into service at some predetermined value of failure. During failure, the latest response can be assigned as the cause for diagnosis.

5. CONCLUSIONS

This paper proposes diagnosis and prognosis of bearing faults using Wireless Modules and Wireless Sensors, where in, a novel two stage neural network has been used to analyze the vibration data obtained from the mechanical test-bed setup. The paper suggests an experimental setup which is a low cost approach for emulating bearing fault in
a motor, to wirelessly sense the vibrations produced due to this fault. The fault classification algorithm used has certain features such as: (a) Using PCA rather than conventional statistical methods to reduce the dimensionality of the data, (b) A method which estimates and recursively calculates the principal components, hence, giving robustness in the diagnosis, (c) Use of Supervised Learning Quantization which has a fast processing time, (d) Finally, Operation Characteristics is plotted and the health index is computed to find the condition of the motor at a given instant. The future work will be to use energy efficient wireless sensors and modules in order to make the Condition Based Maintenance more effective and compare the method with other CBM techniques.

ACKNOWLEDGEMENTS

This work was supported by DoE SBIR through Signal Processing, Inc., 13619 Valley Oak Circle, Rockville, MD20850.

REFERENCES

[24] Lindsay I Smith, “A tutorial on Principal Component Analysis”
[25] Li, Lingxin, “Variable Speed AC Induction motor fault detection and diagnosis using artificial neural networks”, M.Sc.(Eng), Queen's University at Kingston (Canada), 2005, 111 pages; AAT MQ99911


[35] Rezai, Aida “Fault Detection and Diagnosis on the rolling element bearing”, M.A.Sc., Carleton University (Canada), 2007, 130 pages; AAT MR33665


[45] Zhou, Wei, “Incipient bearing fault detection for electric machines using stator current noise cancellation”, Ph.D., Georgia Institute of Technology, 2007, 184 pages; AAT 3294583

**BIOGRAPHY**

**Prasanna Ballal** received his Bachelor of Engineering degree in Electronics and Telecommunication from Mumbai University, India in 2002. He then worked for IndiaGames Ltd., as a software programmer. He received his Master of Science degree in Electrical Engineering in 2005 and Ph.D in Electrical Engineering in 2008 from University of Texas at Arlington, USA. He has been working as a Graduate Research Associate in Distributed Intelligence & Autonomy Laboratory at Automation and Robotics Research Institute, Fort Worth, USA and has authored or co-authored over 14 journal and conference publications. His research interests include Control System Design, Wireless Sensor Networks and Signal Processing.

**Akarsha Ramani** received her Bachelor of Engineering degree in Electrical and Electronics Engineering from Anna University in 2005. She worked with Wipro Technologies as Project Engineer before pursuing her Master of Science degree in Electrical Engineering at the University of Texas at Arlington in 2006. She was a Graduate Research Assistant at Automation and Robotics Research Institute during her Master’s program.

**Matt Middleton** graduated with a B.S.F.S. in International Security Studies from Georgetown University in 2005. Thereafter, he worked as an analyst for Science Applications International Corporation in Northern Virginia. He returned to Texas to pursue a Master of Science degree in Computer Science from the University of Texas at Arlington and is working as a Graduate Research Assistant at the Automation and Robotics Research Institute in Ft. Worth, TX. He is currently researching trust based and discrete event control algorithms for control of autonomous vehicles.
Christopher McMurrough received his Bachelor of Science degree in Computer Science Engineering from The University of Texas at Arlington in 2008. He has been a research assistant in the Advanced Controls and Sensors Laboratory at the Automation and Robotics Research Institute in Fort Worth, Texas since 2006. He is currently pursuing a Masters of Science degree in Computer Science Engineering from the University of Texas at Arlington. His research interests include Autonomous Robotics and Embedded Systems.

Abedalgany Athamneh worked as Protection Engineer at National Electric Power Co. (NEPCO/ JORDAN) for 18 months, and at Yaroumouk University as Lab and Tutorial Instructor for four years. Currently, he is a PhD student at University of Texas at Arlington, looking forward to having more achievements in research at the different fields of the electrical power engineering.

Wei-Jen Lee (S’85-M’85-SM’97-F’07) received the B.S. and M.S. degrees from National Taiwan University, Taipei, Taiwan, R.O.C., and the Ph.D. degree from the University of Texas, Arlington, in 1978, 1980, and 1985, respectively, all in Electrical Engineering. In 1985, he joined the University of Texas, Arlington, where he is currently a professor of the Electrical Engineering Department and the director of the Energy Systems Research Center. Prof. Lee has been involved in research on renewable energy, power flow, transient and dynamic stability, voltage stability, short circuits, relay coordination, power quality analysis, and deregulation for utility companies. He has served as the primary investigator (PI) or Co-PI of over seventy funded research projects totaling over 6.5 million dollars. He has published more than forty journal papers and more than one hundred conference papers. He has refereed numerous technical papers for IEEE, IEE, and other professional organizations. Prof. Lee is a Fellow of IEEE and registered Professional Engineer in the State of Texas.

F.L. Lewis, Fellow IEEE, Fellow IFAC, Fellow U.K. Institute of Measurement & Control, PE Texas, U.K. Chartered Engineer, is Distinguished Scholar Professor and Moncrief-O’Donnell Chair at University of Texas at Arlington’s Automation & Robotics Research Institute. He obtained the Bachelor’s Degree in Physics/EE and the MSE at Rice University, the MS in Aeronautical Engineering from Univ. W. Florida, and the Ph.D. at Ga. Tech. He works in feedback control, intelligent systems, and sensor networks. He is author of 5 U.S. patents, 186 journal papers, 305 conference papers, and 12 books. He received the Fulbright Research Award, NSF Research Initiation Grant, and ASEE Terman Award. Received Outstanding Service Award from Dallas IEEE Section, selected as Engineer of the year by Ft. Worth IEEE Section. Listed in Ft. Worth Business Press Top 200 Leaders in Manufacturing. He was appointed to the NAE Committee on Space Station in 1995. He is an elected Guest Consulting Professor at both South China University of Technology and Shanghai Jiao Tong University. Founding Member of the Board of Governors of the Mediterranean Control Association. Helped win the IEEE Control Systems Society Best Chapter Award (as Founding Chairman of DFW
Chapter), the National Sigma Xi Award for Outstanding Chapter (as President of UTA Chapter), and the US SBA Tibbets Award in 1996 (as Director of ARRI’s SBIR Program).