APPROVAL SHEET

Title of Thesis: Wooden Pole Inspection with a Neural Network Approach

Name of Candidate: Yizhou Wu
Master of, 2016

Thesis and Abstract Approved: ____________Weidong Zhu_____________
(Weidong Zhu)
(Professor)
(Mechanical Engineering Department)

Date Approved: ______11/30/2016___________________

ABSTRACT

Title of Thesis: WOODEN POLE INSPECTION WITH A NEURAL NETWORK APPROACH

Yizhou Wu, Master of Science, 2016

Directed By: Professor Weidong Zhu
Department of Mechanical Engineering

Wooden poles are one of the most commonly used utility carriers in North America. Even though they are under different protection treatments, after decades of weatherworn, wooden poles might have defection because of four main factors: Oxygen, Moisture, Temperature, and PH level, which are suitable for worm growth. Since maintenance and replacement of wooden poles are mainly based on their damage inspection results, an accurate and effective damaging inspection method for wooden poles is essential. However, accuracy of the damage inspection method used by the Baltimore Gas and Electric Company is highly dependent on technicians' experience and inspections always cause extra damage to wooden poles. In this work, an accurate and effective vibration-based wooden pole inspection method is developed in conjunction with a neural network approach. Since the current inspection method is vibration-based, it would not cause any damage to wooden poles during inspection. Lab testing is first conducted using wood samples to verify feasibility of the current inspection method, and two vibration-measurement approaches, which use a microphone and accelerometers, are used to obtain data for neural network analysis. Results from the neural network shows that the current

method can accurately and effectively identify healthy and damaged wood samples. Field testing is then conducted for real wooden poles. Due to complex environment background noise, data from the microphone are no longer effective for neural network analysis and only those from accelerometers are obtained and analyzed using the neural network approach. One hundred wooden poles are tested and final results show that the current vibration-based wooden pole inspection method with the neural network is accurate and effective.

WOODEN POLE INSPECTION WITH A NEURAL NETWORK APPROACH

By

Yizhou Wu

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, Baltimore County, in partial fulfillment
Of the requirements for the degree of
Master of Science
2016

Advisory Committee:
Dr. Weidong Zhu, Advisor
Dr. James Lo
Dr. Meilin Yu

Acknowledgments

I would like to thank my advisor, Dr. Weidong Zhu for his support on my graduate study, Dr. James Lo and Dr. Meilin Yu for agreeing to serve on my thesis defense committee and for contributing their precious time reviewing my manuscript. I would also like to thank all my lab mates for their assistances on the experiment.

Table of Contents

Chapter 1: Introduction

Wooden utility pole is widely used among all utility. According to Baltimore Gas and Electric company (BGE), wooden utility poles are widely used in North America， it can be traced back for over a hundred years; because of its strong structure strength and comparatively low maintenance costs. Southern Yellow Pine is one of major types utilized by BGE. However, after decades of weatherworn, even though these poles were under anti-decay protection [1], wooden poles are still facing erosion threat. In year 2016, there are 38,500 poles required reliability inspection. During inspection, all the poles should be examined by holes-drilling and sample extraction at base to check conditions of decaying, shown in Figure 1.1. There are two holes are randomly drilled on different heights, from close to ground height to men's height. Technicians apply a steel bar and insert into sampling holes to judge the remaining radius of hard wood. Meanwhile, Condition of surface decay is tested by visual examination, around twenty inches from underground. Comparison is made to the percentage of healthy part to remain and decay condition from underground. A final score of specific wooden pole will be generated, and according to that result, different procedures would be applied. However, this inspection method has some shortages; the accuracy of traditional inspection method depends on randomly located drilling position. More importantly, the current inspection method does damage to the pole structure. This method would directly affect pole's life time. This is the reason why a non-damage, conventional inspection method is ideal for wooden pole inspection.

Figure 1.1 Non-conventional Inspection Method

There are many types of conventional inspection means. An ultrasonic tomography technology inspection method for wooden poles developed by a Japanese professor Dr. Tomikawa [2], the idea was based on conventional X-Ray to measure data; on each section layer; the image of each section shows rotten area and healthy wood area with particular simplicity. By gathering multiple section layers, an entire X-ray image for a particular wooden pole can be captured. However, the defect for ultrasonic inspection method is that ultrasound doesn't propagate linearly because of Yong's modules is different in sap and center. Moreover, ultrasound can't penetrate rotten area. To clarify, if ultrasonic meets a pocket, no further detection can be reached to the core of a wooden pole, thus it can't have a complete of a certain section. Wooden pole inspection is not stopped only by ultrasonic, Dr. Wyckhuyse and Dr. Maldague [3, 4] approach to infrared thermography to inspect wooden poles. Infrared thermography technology built a model based on different moisture content to the sound wood, in order to compare of wood thermal properties.

The work is focused on bringing practical inspection means to technicians, making it effective and time-saving. Other inspection method, like ultrasonic and infrared thermography and have its own limits for applications. Ultrasonic technology scans each chasm of an entire pole. Although the image of decay would be clear, the device takes time to analysis the image, requiring technician to measure the entire pole with this scanning procedure. A more practical scenario, a technician inspects 50 poles in 8 hours working time; convert to each pole for 10 minutes. Additionally, signal of infrared thermography would be retarded by exterior wood, without a chance to detect any decay interior or core of the pole, unless decaying position is close to the surface of a pole.

Many other studies on wooden pole inspections have been made similar to inspection methods introduced above. Although their advantages are quite obvious, the shortages are also quite noticeable. As for Microphone method, Dr. Sabatier [5] published a paper about Microphone for soil physical properties. Dr. Sabatier focus on evaluating soil physical properties by gathering acoustic sound and phase by inverting sound signal into two porous materials in capillary tube model. Dr. McGraw [6] introduces a phosphorescence microphone to test oxygen sensors and films in detailed explanation in her paper on this procedure.

No matter what kinds of inspection methods are applied, we should take all advantages as well as reduce defects. Most importantly, all means should match engineering practical needs for inspection, such as time efficiency and easy application. In order to meet engineering practical concerns for inspection and accuracy of detecting decay and pocket, this paper approaches Neural Network modeling.

Neural network modeling approximates the idea of human brain structure~~.~~ Neural network modeling propagates weights from one layer to the next, ~~like~~ similar to neuron to neuron. A neural network involves large amounts of data and the relationship among the data. Initially fed with a trained data model~~;~~ the input enters values to neural network model, from which an output will be generated, to the most accurate results. Usually, the more input data for the neural network model, the more accurate the model will be.

Neural network uses several principles, such as fuzzy logic, winner takes all, perception, feed-forward and so on. Wooden pole has a nonlinear feature. The approach is superior to the approach using recurrent neural network (RNN) in many ways. The RNN approach applies detection and localization of damages in any objects in which the sound and vibration propagation is linear. On the other hand, as a feed-forward neural network with multiplayer nonlinear perception, back-propagation neural network (BPNN) is applicable for modeling of wooden poles. It is known that BPNN method typically fits for nonlinear structure system like tree barks and structure with nodes. BPNN and RNN are the typical learning method suitable for nonlinear case.

This paper will focus on two kinds of neural network training methods: BPNN and RNN; two kinds of training methods will be compared with each other.

BPNN is a very classical approach to non-linear problems. Among all Neural Network, BPNN is the best way to approach multi-layer nonlinear material. This is typical of the inside of a wooden structure. All trees, including Southern Yellow Pine have nonlinear structure, when erosion begins in the pole, the structure inside consists of several small pockets, making it more non-linear. Our idea of applying BPNN is to study the structure inside, by imaging the wooden inside ~~is~~ in one spot after ~~a~~ another spot. When two spot connect~~ed~~, the information would transmit to the next spot. If the two spot do not connect, that information couldn't be transmitted. Instead, modified information would eventually get to the specific spot that doesn't connection with previous one. Although through another spot's connection, it would receive. But this information is not correct. When building ~~this~~ a kind of structure, we would introduce "Weight". Weight is used to measure how much difference from the spot it should have. For example, if a line is on a sine wave, and in the line, there is a weak connection, eight would compensate for that value. For eight in a non-linear multi-layer case each spot has its own weight. By ~~c~~alculating weight and input value, the output is the result.

In the scope of testing chapter, two kinds of testing equipment will be introduced. One is microphone vibration sound testing, the other is acceleration transducer testing. Both are broadly used in field testing. Each of these testing methods has its own advantage and disadvantage. For example, microphone testing directly obtains feedback sound after hammer knocking the pole itself. Feedback is straightforward and easy to interpret~~.~~ As for disadvantage, it requires relatively higher background noise. Each single time testing by microphone would be affected by airplane noise, bird singing, even a vehicle horn. On the other hand, acceleration transducers require a fine attachment surface. Its signal may not reflect a real situation. Because tree~~s~~ surfaces are not plane, it is difficult to attach transducers on the tree.

At the experiment phase, the whole testing plan is consisted of several parts. In the first phase, prove the data extracted from vibration and transducer can be applied to neural network. During first phase, examiner needs to find the most suitable neural network model. The second step is to build a math model. Fortunately, only adapt Mat-lab neural network model to execute calculation to prove math model is correct. The second phase requires Lab testing. The supplier from BGE provides 10 wooden samples obtained from field. In review of those samples, different conditions of erosion and whether it has pocket occur. We classified 3 categories. Categories are good, slightly eroded and pocket. In this phase, we build a new model of BPNN in order to find a more accurate model for field testing in next phase; it also doing this brings us more accurate data, because those pole samples are obtained from real world after several decades of seasonal weather-beaten exposure. The pockets and conditions of erosion that we can conduct in lab are equal to field situation. The third phase is field testing. In this step, examiners will go to field, extract data directly from in-use wooden pole utilities. The most difficult part is to test the previous model and whether it can detect erosion and pocket from below the earth. In fact, the under-earth part of pole is most likely to have erosion and pocket situations. Therefore, whether the whole inspection plan can work, depends on field testing. In order to conduct this field testing, equipment required are Portable Batteries, Converter, Microphone, Siglab, Laptop, Transducers, Hammer.

The remaining part is organized as follows. The kinematics of a three-dimensional Euler–Bernoulli beam is first discussed in Sec. 2. Governing equations of the beam and constraint equations are derived in Sec. 3 using Lagrange's equations for systems with constraints. Static equilibrium and linear dynamics problems are formulated in Sec. 4. Numerical examples are presented in Sec. 5 to demonstrate the performance of the current formulation. Finally, some conclusions from this study are presented in Sec.6.

Chapter 2: Neural Networks Training Method

Artificial Neural Network is broadly used in machine learning. It used to approximate functions that can be adapted on inputs and outputs. Artificial Neural Network has supervised learning and unsupervised learning. In this chapter, BPNN and RNN as supervised learning method are introduced for wooden pole training.

In order to prove damaged wooden pole is non-linear character, a linearity validation experiment is conducted. Shown as Figure 2.1. In engineering convenience, linearity validation is conducted in following method.

1. Randomly mark Point A and Point B on a single wooden pole.
2. Use hammer hitting Point A and accelerometer attaches on Point B.
3. Reverse Point A and Point B, conduct step 2 again.

In Figure 2.1, green curve is from first vibration acquisition, red curve is from second vibration acquisition. Their FRF plot doesn't match each other; peaks are not matching the same frequency. Thus, wooden pole is non-linear.

In Mathematic aspect, linearity verification should be conducted by Amplitudes superposition. That is, normal hard force plus twice hard force should equal to three times hard force. In favor of engineering convenience and practical purpose, nonlinearity experiment is validated through Points exchange.

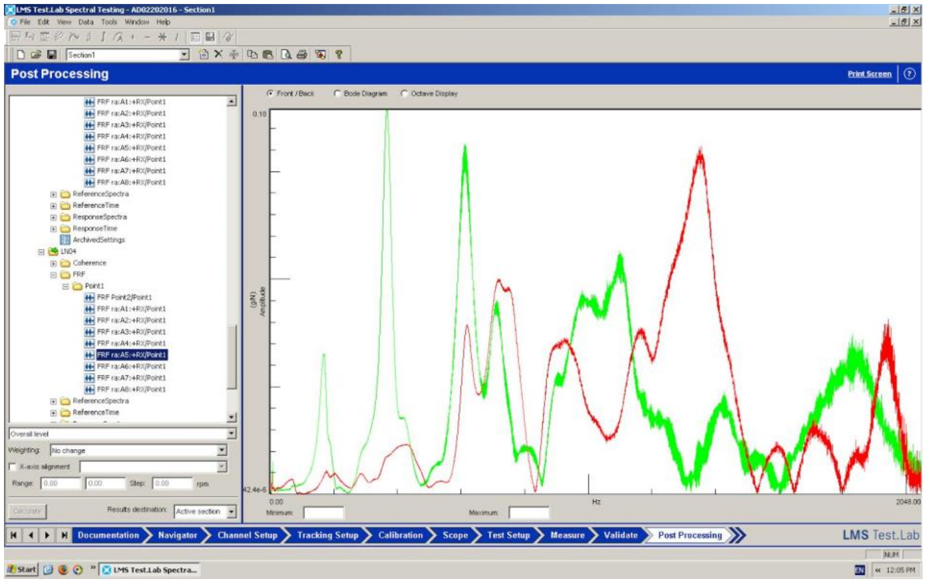


Figure 2.1 Wooden pole non-linearity validations

After wooden pole linearity experiment, data shows wooden pole has strong non-linear character, especially on corrosive poles. BPNN and RNN can be applied for this situation. In the following, BPNN and RNN training method will be explained. Each method will be compared with each other, to decide which one could best match the requirement for the inspection.

2.1 Back Propagation Feed- Forward Neural Network

Feed-forward and Back-Propagation are the two major steps for BPNN [7]. In BPNN, it has three layers: Input layer, hidden layer and output layer. Because BPNN is a supervised learning method, desired outputs are required in training. The aim for BPNN is: Calculate the gradients of loss function with respect of weights and networks.

In feed-forward step, apply the input values through layers with suitable propagates, that's weights, with bias all the way to the desired outputs. In Back-

Propagation phase, the algorithm chases the minimum value of error function. It traces back to adjust the weight values by each neuron. Along with several iterations, finally the model reaches stable, the error value between Target and output reaches a minimum.

<u>Mathematical approach</u>

Observing from wooden pole structure, it consists of multilayers of year rings. Although it has layers, the layer is not linearized. Therefore, the capability of computing in a wide range of judging function is suitable for multilayered networks. Therefore, in order to handle a black box model with hidden nodes and unknown number of neurons and of its structure in a multilayer case, Back-propagation algorithm will be introduced.

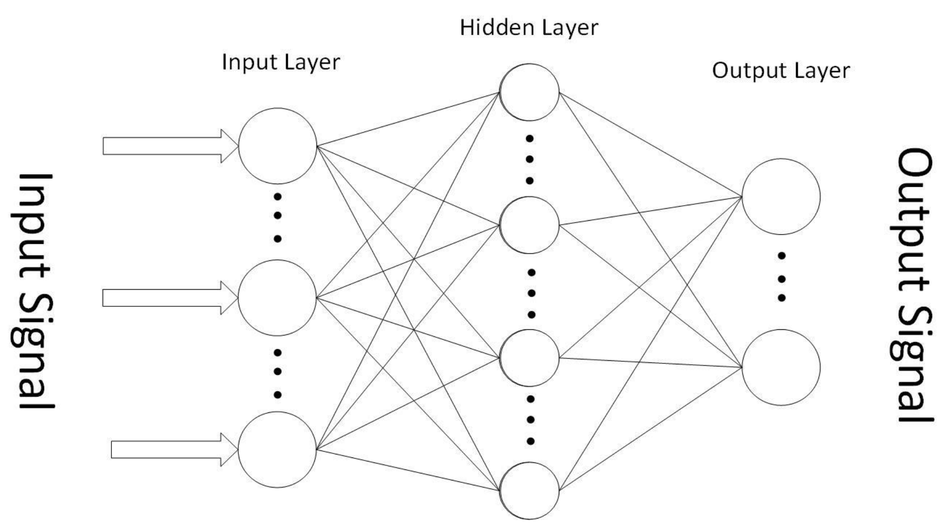


Figure 2.2 Back-Propagation structure layouts

Back-Propagation algorithm adapts supervised learning, uses data from inputs as well as desired outputs to train the math model. Once training begins, outputs compute with new inputs, and it varies from time to time. The only value won't change is the weight of each node. Weight is the key to build a model. A simple form of neural network shown as following:

$$A = f(W*P+b) = f(\sum_{j=1}^{r} W_j P_j + b) \qquad (2.1)$$

Where A is output Matrix, $W=[W_1, W_2, \cdots W_r]$ is weight, $P=[P_1, P_2, \cdots P_r]^T$ is inputs matrix, b is bias.

The bias can be easily being told that it is another input value based on W*P. Additionally, bias is also a weight value. Normally bias is settled as 1 for input. In the network design phase, bias is also very important, as it increases the possibility for solving problems by letting diagrams of activation functions move around.

Activation functions

Gradient descent is the crucial point introduced in activation function. Back-Propagation use error functions as reference value of weight. The error function will be minimized after combination by using series of combination of weights and neurons feedforward calculation. The error functions are also important in calculation at each step of iteration. During each iteration step, like other iteration functions, make sure those error functions are continuous and differentiable. Moreover, steps are not to input to error function, because not similar to activation function, step function is discrete. Here, activation function match error functions requirement: continuity and differentiable. The descent of gradient of a sample activation function derives as Eq. (2.2)

$$S_c(x) = \frac{1}{1+e^{-cx}} \qquad (2.2)$$

Where $1/c$ is associate with temp in neural network, it is randomly generated number. The shape of sigmoid activation function would be also changed, When c changes. Similarly, the value of sigmoid function will be closer to step function, once c increases, shown as Figure 2.2.

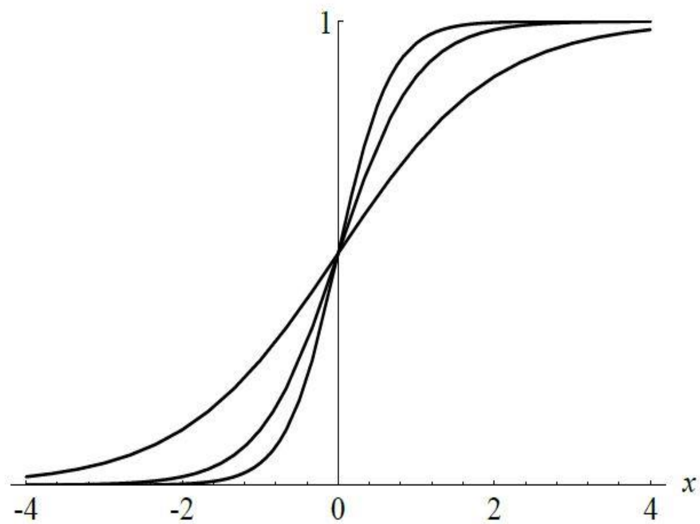


Figure 2.3 Shape of sigmoid activation function

The derivative function of the sigmoid activation functions as:

$$\frac{d}{dx}s(x) = \frac{e^{-x}}{(1+e^{-x})^2} = s(x)(1-s(x)) \tag{2.3}$$

Derive for the sigmoid activation function to get symmetrical function $S(x)$:

$$S(x) = 2s(x) - 1 = \frac{1-e^{-x}}{1+e^{-x}} \tag{2.4}$$

The $S(x)$ is a hyperbolic tangent function. Tangent function and sigmoid function are the two most popular functions used for Back-propagation algorithm. User can decide to use one each that matches its requirement.

Error function calculation

At a node n, the activation value O_n, the Target value is T_n, the error function is

$$E = \frac{1}{2}\sum_{n=1}^{P}\left\| T_n - O_n \right\|^2 \tag{2.5}$$

After calculating error function, a minimized gradient will be generated for this training set, this value measures gap between target and output. The network would judge the gradient values after several iterations of weights adjustment whether the gradient of error function reaches a minimum.

Approaching steps

Steps 1-6 are the feed-forward steps, since step 7 and the following are the Back-Propagation steps.

Step 1: Calculate the total for hidden neuron.

$$Net_{hi} = \sum_{j=1}^{n} W_{ij} * P_j + b_i \tag{2.6}$$

There, W_{ij} is the weight parameter for connection key between neurons, P_j is the input for each neuron, b_j is the bias value to compensate for current layer.

Step 2: Then apply activation function

$$Out_{hi} = \frac{1}{1+e^{-net_{hi}}} \tag{2.7}$$

Carry the Out_{hi} out for the same process for the same layer neuron; repeat the same process for all hidden layers of neuron.

Step 3: Then comes to output.

$$Net_{oi} = \sum_{j=1}^{n} W_{ij} * Out_{hj} + b_i \tag{2.8}$$

There, W_{ij} is the weight parameter for connection keys in hidden layer, Out_{hj} is the neuron output of the first input layer, b_i is the bias value to compensate for hidden layer .

Step 4: Apply Activation function again

$$Out_{oi} = \frac{1}{1 + e^{-Net_{oi}}} \tag{2.9}$$

Step 5: separately estimate outputs to target values for the error function

$$E_i = \frac{1}{2}(T\arg et_{oi} - Out_{oi})^2 \tag{2.10}$$

Step 6: Total error for the neural network is the sum of errors

$$E_{total} = \sum_{i=1}^{i} E_{oi} \tag{2.11}$$

Then, one would get the value from $\frac{\partial E_{total}}{\partial W_1}$,

$$W_{ij}^{+} = W_{ij} - \eta * \frac{\partial E_{total}}{\partial W_{ij}} \tag{2.12}$$

And

$$W_{ij}^{+} = W_{ij} - \eta * \frac{\partial E_{total}}{\partial W_{ij}} \tag{2.13}$$

Update the value from each W_i^{+} , after several iterations, system would reach a regression status. A BPNN neural network model will be created.

BPNN Examples

Following case shows a BPNN application example of mathematical calculation. [8]. For ease of understanding, eliminate bias and downsizing to two inputs and one hidden layer.

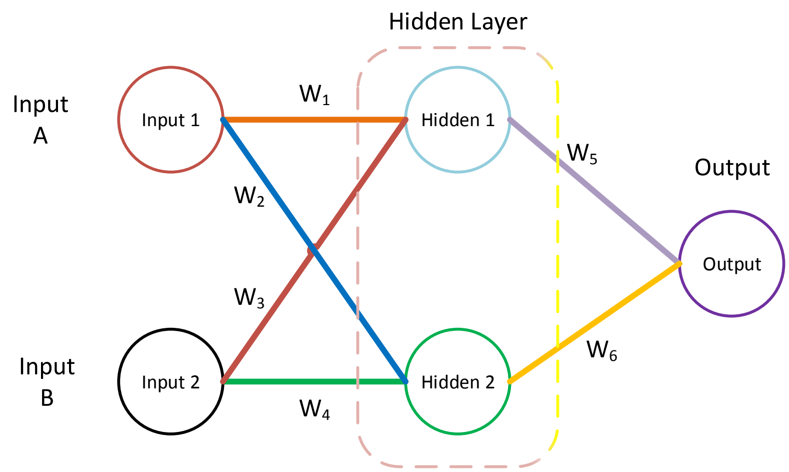


Figure 2.4 BPNN layouts with weights

In order to calculate error function, gives each cell an initial value:

$$\begin{aligned}&A=0.4, \quad W_1=0.1, \quad W_3=0.4, \quad W_5=0.2,\\&B=0.9, \quad W_2=0.7, \quad W_4=0.6, \quad W_6=0.8,\\&Target= \ 0.5,\end{aligned} \tag{2.14}$$

According to Eq. (2.6), one has

$$Net_{h1}=A*W_1+B*W_3 \tag{2.15}$$

Substituting parameters in Eq. (2.14) into Eq. (2.15) yields

$$Net_{h1}=0.4*0.1+0.9*0.4=0.4 \tag{2.16}$$

According to Eq. (2.6), one has

$$Net_{h2}=A*W_2+B*W_4 \tag{2.17}$$

Substituting parameters from Eq. (2.14) into Eq. (2.17) yields

$$Net_{h2} = 0.4*0.7 + 0.9*0.6 = 0.82 \tag{2.18}$$

According to Eq. (2.9), sigmoid activation function is

$$Out_{h1} = \frac{1}{1+e^{-net_{h1}}} \tag{2.19}$$

Substituting Eq. (2.16) into Eq. (2.19) yields

$$Out_{h1} = \frac{1}{1+e^{-0.4}} = 0.60 \tag{2.20}$$

Substituting Eq. (2.18) into Eq. (2.19) yields

$$Out_{h2} = \frac{1}{1+e^{-0.82}} = 0.69 \tag{2.21}$$

According to Eq. (2.22), output is

$$Output = Out_{h1}*W_5 + Out_{h2}*W_6 \tag{2.23}$$

Substituting Parameters from Eq.(2.20), Eq.(2.14) and Eq.(2.21) into Eq.(2.23) yields

$$Output = 0.6*0.2 + 0.69*0.8 = 0.67 \tag{2.24}$$

According to Eq.(2.10), Error Function is

$$E = \frac{1}{2}(Target - Output)^2 \tag{2.25}$$

Substituting Parameters from Eq. (2.25)and Eq. (2.24)into Eq. (2.25)yields

$$E = \frac{1}{2}(0.5 - 0.67)^2 = 0.01445 \tag{2.26}$$

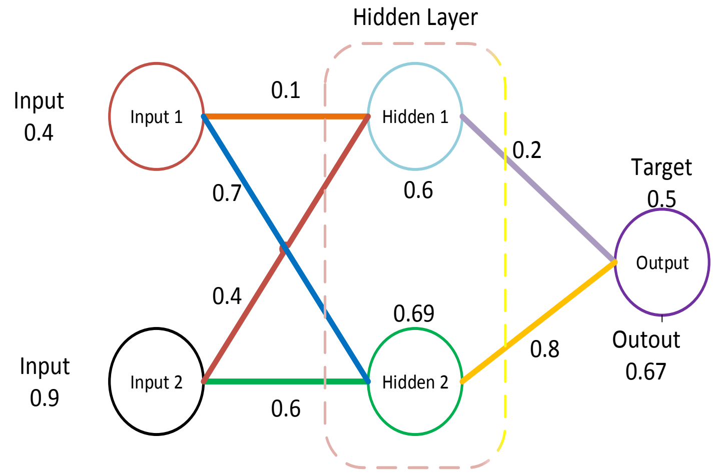


Figure 2.5 BPNN layouts with given weights

In order to minimize the value of error function, a Back-propagation method is applied to check each the value of gradient of descent $\frac{\partial E}{\partial W}$, then update weight values on each chain all the way back to the very first layer.

According to Eq.**Error! Reference source not found.**, one has

$$\frac{\partial E}{\partial W_5} = -(T\arg et - Output) * \frac{\partial Output}{\partial W_5} \tag{2.27}$$

Substituting parameters of Eq.(2.27) yields

$$\frac{\partial E}{\partial W_5} = 0.0226 \tag{2.28}$$

According to Eq.**Error! Reference source not found.**, one has

$$\frac{\partial E}{\partial W_6} = -(T\arg et - Output) * \frac{\partial Output}{\partial W_6} \tag{2.29}$$

Substituting parameters from Eq. (2.29)yields

$$\frac{\partial E}{\partial W_6} = 0.0260 \tag{2.30}$$

Then each weight's value needs to be updated. According to Eq.(2.12), one has

$$W_5^+ = W_5 - \frac{\partial E}{\partial Out_{h1}} = 0.2 - 0.0226 = 0.1774 \tag{2.31}$$

According to Eq.(2.12), one has

$$W_6^+ = W_6 - \frac{\partial E}{\partial Out_{h2}} = 0.8 - 0.0260 = 0.774 \tag{2.32}$$

According to the same procedure of Eq.(2.27), one has

$$\frac{\partial E}{\partial W_1} = \frac{\partial E}{\partial (Target - Output)} * \frac{\partial (Target - Output)}{\partial Output} * \frac{\partial Output}{\partial Net_{h1}} * \frac{\partial Net_{h1}}{\partial W_1} \tag{2.33}$$

The serious solution of descent of gradient 1~4 are listed as Eq.(2.33)

$$\begin{aligned} &\frac{\partial E}{\partial W_1} = 6.4*10^{-4}, \quad \frac{\partial E}{\partial W_2} = 2.49*10^{-3}, \\ &\frac{\partial E}{\partial W_3} = 1.44*10^{-3}, \quad \frac{\partial E}{\partial W_4} = 5.6*10^{-3} \end{aligned} \tag{2.34}$$

Substituting parameters from Eq.(2.31) and Eq.(2.32) yields

$$\begin{aligned} &W_1^+ = W_1 - \frac{\partial E}{\partial W_1} = 0.09936, \quad W_2^+ = W_2 - \frac{\partial E}{\partial W_2} = 0.69751 \\ &W_3^+ = W_3 - \frac{\partial E}{\partial W_3} = 0.39856, \quad W_4^+ = W_4 - \frac{\partial E}{\partial W_4} = 0.5944 \end{aligned} \tag{2.35}$$

Substituting parameters from Eq.(2.35) and Eq.(2.14) into Eq.(2.36) yields

$$Net^+{}_{h1} = A*W_1^+ + B*W_3^+ = 0.3984 \tag{2.36}$$

Substituting parameters from Eq.(2.35) and Eq.(2.14) into Eq. (2.37)yields

$$Net^+{}_{h2} = A*W_2^+ + B*W_4^+ = 0.813964 \tag{2.37}$$

Substituting parameters from Eq. (2.38) into Eq. (2.39) yields

$$Out_{h1}^+ = \frac{1}{1 + e^{-Net_{h1}^+}} = 0.59830 \tag{2.40}$$

Substituting parameters from Eq.(2.40) into Eq. (2.41)yields

$$Out_{h2}^{+} = \frac{1}{1+e^{-Net_{h2}^{+}}} = 0.69295 \qquad (2.41)$$

Substituting parameters from Eq. (2.23)into Eq. (2.42)yields

$$Output^{+} = W_5^{+} * Out_{h1}^{+} + W_6^{+} * Out_{h2}^{+} = 0.6425 \qquad (2.42)$$

Substituting parameters from Eq.(2.42) into Eq.(2.43) yields

$$E^{+} = \frac{1}{2}(T\arg et - Output^{+})^2 = 0.5*(0.5-0.6425)^2 = 0.0102 \qquad (2.43)$$

Shown as the result of error function, E=0.01445> E^{+} =0.0102, by using gradient descent actually improves training results. After more iterations, the error will reduce. Thus, this is how the BPNN training method works.

2.2 Recurrent Neural Network

Recurrent neural network was developed in the 1980s. Compared to other developed training methods in neural network, it is a relatively a new approach. It includes many possibility, and reviews are now available for RNN, this includes [9]. The variable in life, such as: figures recognition, feed-forward thinking meets requirement. However, RNN is unit architecture; has a feedback loop to update the weight value. It is a supervised learning method; scientists always choose to use discrete time setting to training for sequences.

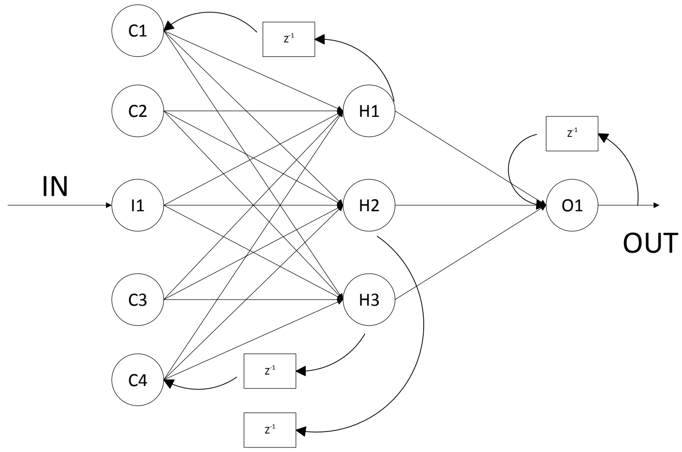


Figure 2.6 Recurrent Neural Network Layouts

Mathematical approach

The previous chapter introduced BPNN. Accordingly, this chapter will introduce a recurrent training method. By contrast to Feed-Forward Method, recurrent neural network has some advantages like:

- Recurrent neural network is much closer to real biological neurons, because all the biological neural are recurrent.
- There is no definite winner in this model, it has several training algorithms.
- Recurrent neural network plays in dynamical system.
- Recurrent neural network is a supervised training method, so it can approximate arbitrary with arbitrary precision.
- It’s not a popular training method.

Because recurrent neural network is like a black-box for engineering, it fits for different types of signal processing.

Supervised Training

In Recurrent Neural Network training [10] has two methods. It has supervised and un-supervised, in order to recognized the pattern and towards what consequence is desired, here will introduce the supervised one.

A whole scale of recurrent neural network is consisted by inputs, hidden layers, and output units~~.~~ Each layer has a certain number of neurons and they are connected by synaptic strength. At a given situation, no matter inputs, outputs and hidden layers are called units. In each unit, there is activation. Activation is denoted separately by inputs $u(n)$, output units by $y(n)$, internal units by $x(n)$. In some situations, those separated parts are ignored, just by adapting $x(n)$. The formula

$$x_j(n+1) = f(\sum W_{ij} X_j(n)) \tag{2.44}$$

is a discrete time of RNN; and

$$\Gamma \dot{x}_i = -x_i + \sum W_{ij} f(x_j) \tag{2.45}$$

is a continuous time of RNN.

The solution of RNN is to approach the expended algorithm by piling up the identical sizes of copies of RNN. The connection will be re-directed to the network to gain chains of connections among those copies.

Weights of each unit, w_{ij}^{in} , w_{ij} , w_{ij}^{out} , w_{ij}^{back} are identical between them. Therefore, the training data connect a line of input to output time series

$$u(n) = (u_1(n), \ldots\ldots u_k(n))' \tag{2.46}$$

Then differential the formula above

$$d(n) = (d_1(n), \ldots\ldots d_L(n))' \tag{2.47}$$

Beginning from the very first epoch to each epoch introduced later on, the forward passing is updated the old stacked network, because of recurrent character. Therefore, at each epoch, the u (n) is read from inputs, however the inertial part x (n) is updated from inputs u (n) and also affected by previous epoch x (n-1). As more and more epochs from hidden layer are computed, the output y (n), is eventually derived.

Error Function

With updates from each time, the value of outputs are updated, therefore the value of Error Function is going to be minimized.

$$E = \sum_{n=1,\dots,T} \| d(n) - y(n) \|^2 = \sum_{n=1,\dots,T} E(n) \qquad (2.48)$$

The aim of every neural network is to minimize the value of error function. Once the value is smaller, the shape after training is much closer to the real d (n).

Computation Steps

In RNN training, included are three steps, Forward Pass, Compute by Backward, and adjust the connection weights.

Forward Pass

By applying formula 3.3, 3.4, forward pass consists of each updated neuron of each epoch. It starts from the very first epoch, and then goes through all the epochs in hidden layers to the final output. From the description, one can tell the forward pass doesn't have much difference from other training methods, like WTA, or BPNN training method. However, the key is the update stack value from each neuron. Those values make a big difference from training model.

Compute by Backward

In this step Time is from 1 to T, but the proceeding is backwards, where n is from T to 1. For each time n, denote the input activation $x_i(n)$, output activation $y_i(n)$, error term for the back- propagation is $\delta_i(n)$

$$\delta_j(T) = (d_j(T) - y_j(T)) \frac{\partial f(u)}{\partial u} |_{u=z_j(T)} \tag{2.49}$$

Error propagation is to measure the error range between target value and actual value. Each unit has its own discrepancy.

For outputs at layer T

$$\delta_i(T) = \left[\sum_{j=1}^{L} \delta_j(T) w_{ji}^{out} \right] \frac{\partial f(u)}{\partial u} |_{u=z_j(n)} \tag{2.50}$$

The same reason for the discrepancy in internal units in layer T, it shows as below:

For internal units $x_i(T)$, at layer T

$$\delta_j(n) = \left[(d_j(n) - y_j(n)) + \sum_{j=1}^{N} \delta_i(n+1) w_{ij}^{back} \right] \frac{\partial f(u)}{\partial u} \Big|_{u=z_j(n)} \tag{2.51}$$

The same reason for the discrepancy in output units, the calculation method is different from previous equation. In the output unit, the accumulated value of n+1 error propagation times with adjusted weight plus the accumulated value of n error propagation time with adjusted output’s weight. In this equation, it shows the characteristic of backward calculation in recurrent method.

For the output units of earlier layers

$$\delta_i(n) = \left[\sum_{j=1}^{N} \delta_j(n+1) w_{ji} + \sum_{j=1}^{N} \delta_j(n) w_{ji}^{out} \right] \frac{\partial f(u)}{\partial u} \Big|_{u=z_i(n)} \tag{2.52}$$

For internal units $x_i(n)$ at earlier times, where $z_i(n)$ again is the potential of the corresponding unit. In order to compute error propagation in backward computation, it needs to know the adjusted weights value in each unit. As shown in the equation above. w_{ji} And w_{ji}^{out} .

Adjust the connection weights

Different from error propagation calculation, weights are computed by iteration. The value at n-1 is displaced by the value at n.

New

$$w_{ij} = w_{ij} + \gamma \sum_{n=1}^{T} \delta_i(n) x_j(n-1) \tag{2.53}$$

where $x_j(n-1) = 0$ for n=1. So, the input unit is followed the same procedure as above iteration.

$$w_{ij}^{in} = w_{ij}^{in} + \gamma \sum_{n=1}^{T} \delta_i(n) u_j(n) \tag{2.54}$$

New

$$w_{ij}^{out} = w_{ij}^{out} + \gamma \begin{cases} \sum_{n=1}^{T} \delta_i(n) u_j(n), \text{if } j \text{ refers to input unit} \\ \sum_{n=1}^{T} \delta_i(n) x_j(n), \text{if } j \text{ refers to hidden unit} \end{cases} \tag{2.55}$$

New

$$w_{ij}^{back} = w_{ij}^{back} + \gamma \sum_{n=1}^{T} \delta_i(n) y_j(n-1) \tag{2.56}$$

where $y_j(n-1) = 0$ for n=1.

2.3 Chapter Summary

It is very clear that BPNN and RNN are quite different RNN updates weights and also weight of hidden neurons for every single iteration epoch. However, the

BPNN only re-calculates the weights and feed forward to the desired value. Because of different structure, RNN may differ to calculate error function; it changes too many parameters to find a right pattern.

Chapter 3: Data processing for wooden utility poles

3.1 Scope of Data Processing

In order to build a mathematical model to tell difference of conditions of healthy or unhealthy, it requires a large number of testing samples. There are over 100 wooden poles tested for the experiment, over 6000 vibration response samples have been collected. Regardless of sample quality or diversity of wooden pole ages, and hitting points of range, data capacity is quite enough to have a complete data analysis. This enables experiment accuracy for the wooden pole inspection experiment. The experiment consists of Microphone acoustic testing and Accelerometer vibration testing comparing data collected from vibration and acoustic approach and extent of model fitting percentage. Eventually, the best testing method will be applied to conduct field testing. Each hammer hitting point is designed to hit 5 times average, 20 times for each wooden pole. Following the standard wooden pole inspection procedure and evaluation standard, the same inspection standard as well as testing results are applied during neural network data processing. In wooden pole inspection cases, the class category from one to four are the actual values set as Target values during neural network training, shown as Table 3.4. The entire testing procedure has two major phases, which are lab testing phase and field testing phase. In lab testing, an executable testing six wooden pole samples vibration and acoustic data will be collected, with adjustments to the practical situation.

Table 3.4 Inspection Score Table

Inspection		Class	Score	Procedure	
Internal	External			Evaluation	Action
healthy wooden	Rotten Wooden				
>0.10 m	without	1	80~90	Good	None
0.07 to 0.10 m	max 0.01 m	2	70~80	Initial decay	Retreat Int./Ext.
0.03 to 0.70 m	max 0.02 m	3	60~70	Advanced decay	Retreat Int.
<0.03 m	total	4	>60	Failure	Replace

3.2 Model Decomposition Algorithm & Lab Testing

In lab testing, testers are supplied with 6 retired wooden utility poles, which are around 2-3 feet long. Due to the volume of samples for lab testing are not compatible with field testing set, the target set for lab testing has a minor change. According to observation for internal and external decay conditions and actual feeling measured by hands, the 6 poles are separated into 2 classes, which are healthy and unhealthy. From Figure 3.1, those in healthy conditions are in the first row, those suffered with decay and pocket are in the second row.

The 6 wooden utility poles are separately tested by microphone acoustic tests and accelerometer vibration tests. During sensors setup, each wooden utility pole is attached 8 accelerometers and marked evenly with 20 points for hammer hitting. 4 accelerometers are placed 90 degrees each around poles top circle surface, another 4 accelerometers are placed 90 degrees each, around the poles bottom circle surface; microphone is placed 30 cm away, 40 cm high from wooden utility pole. Every pole is marked evenly with 20 points for hammer hitting, Shown as Figure 3.2.

(a) (b) (c)

(d) (e) (f)

Figure 3.1 Six wooden utility poles for lab testing

All wooden pole samples for this lab testing are southern pine, they have the same features but different ages, and species difference can be ignored at this time. As Figure 3.1 shows, the different conditions of wooden pole samples, Figure (a), (b), (c) are in better condition with no decaying or pocket exist inside or the surface of poles. On the contrary, Figure (d), (e), (f) are the unhealthy condition of wooden pole samples, with decaying on the surface and Pocket inside the pole.

Figure 3.2 Sensors and microphone placement

All six wooden pole samples are standing exactly like Figure 3.2, during testing, room background sound level is less than 10 dB, after listening to each clip, no background noise has impact to test results, and sound clips are acceptable for further analysis. Meanwhile, room environment temperature remains at 17 to 20 ℃， testing setup are the same for samples.

3.2.1 Accelerometer vibration test

Data is collected respectively based on vibration test. 960 set of data though 8 accelerometers is listed in Appendix A. In order to eliminate the impact from hammer and force/impact difference during lab test, adapt FRF measurements with best correlation for each hammer hitting. FRF testing generates response data. In Frequency domain, the testing range is between (0~ 2400) Hz. To convert the amplitude, here one uses normalized amplitude, which is actual amplitude divided by largest amplitude. The display range is from (0~1). Figure 3.3 and Figure 3.4 list a single hammer hit response of 8 accelerometer sensors of a single pole. From Figure 3.3, curves have a great repeatability, tend to be more uniform; contrarily, from Figure 3.4, curves tend to be more randomly generated. That is, natural frequencies of healthy poles locate in a small area, and have less variance compare to unhealthy poles. From this point, collect the natural frequencies from curves and put natural frequencies into training is a way of approaching. From Figure 3.3 and 3.4, the normalized amplitude doesn’t show a strong connection to classification for healthy and unhealthy poles, even useless for cutoff frequencies.

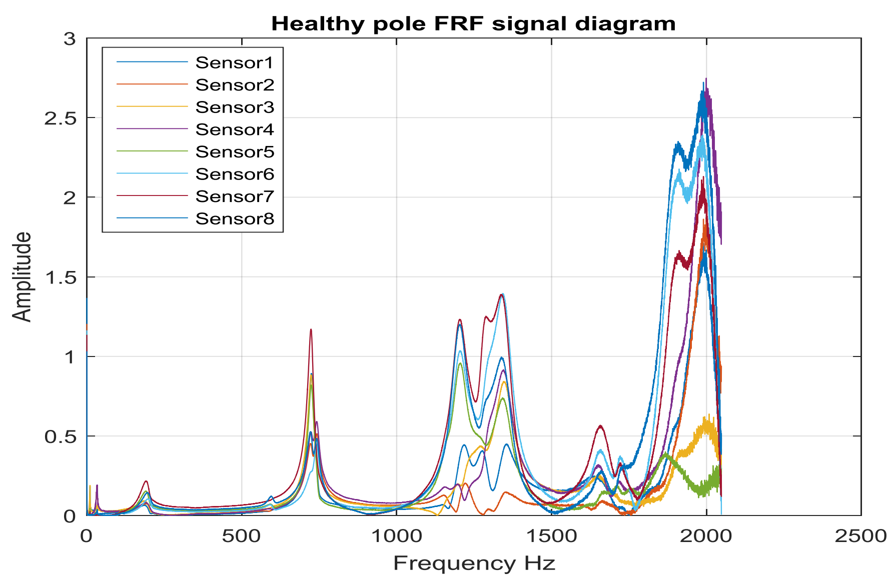


Figure 3.3 FRF response of healthy pole

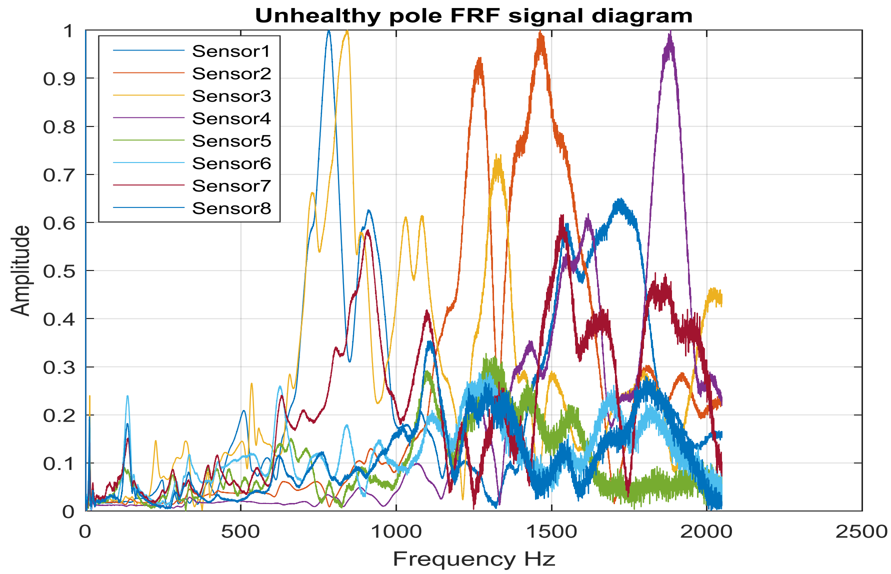


Figure 3.4 FRF response of unhealthy pole

From Figure3.3, the response of accelerometers of a single hit shows strong uniformness; On the contrary, response of Figure3.4 doesn't show uniformness. However, on the other perspective, one accelerometer with multiple hits holds the same results as expected. Figure3. 5 and Figure3.6 illustrate hammer hitting point doesn't have impact on frequency of a single sensor.

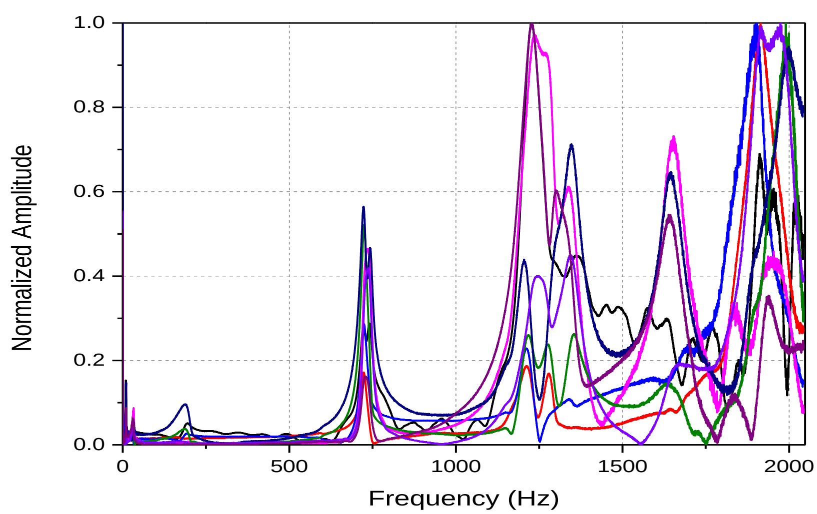


Figure 3.5 Healthy pole FRF response a single sensor for multiple hits

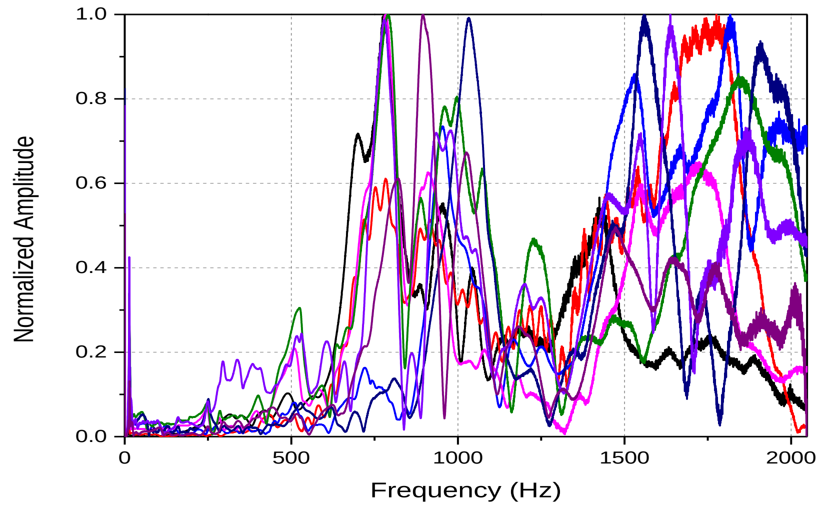


Figure 3.6 Unhealthy pole FRF responses a single sensor for multiple hits

Therefore, results from two major factors can be concluded from Figure3.3, 3.4, 3.5, and 3.6. First, FRF testing response reflects that different accelerometers at different position on a single wooden pole reflect differently between a healthy wooden pole and an unhealthy pole. Anywhere on a single healthy pole should be the same natural frequencies; however, the natural frequencies from a single unhealthy pole are different from measuring points. This is true because wood has nonlinear characteristic. Second, the same sensor at various spots on a single wooden pole reflects differently between a healthy wooden pole and an unhealthy pole. From Figure 3.5 and Figure 3.6, no matter where to put the sensor, the FRF response should remain the same, so is the natural frequencies.

Based on those two factors, finding the locations of natural frequencies are a key to tell difference of between healthy and unhealthy wooden poles; while peaks are not a key at this point.

From Figures above, curves combine with strong noisy signal; it would affect peaks extracting accuracy when one faces large volumes of data. Thus, before extracting peaks, it is important to smooth signal. In the scope of data processing, various kinds of tool can be used. High pass filter and low-pass filter and so on. A low-pass filtering [5] is the result after filtering shown as Figure 3.7

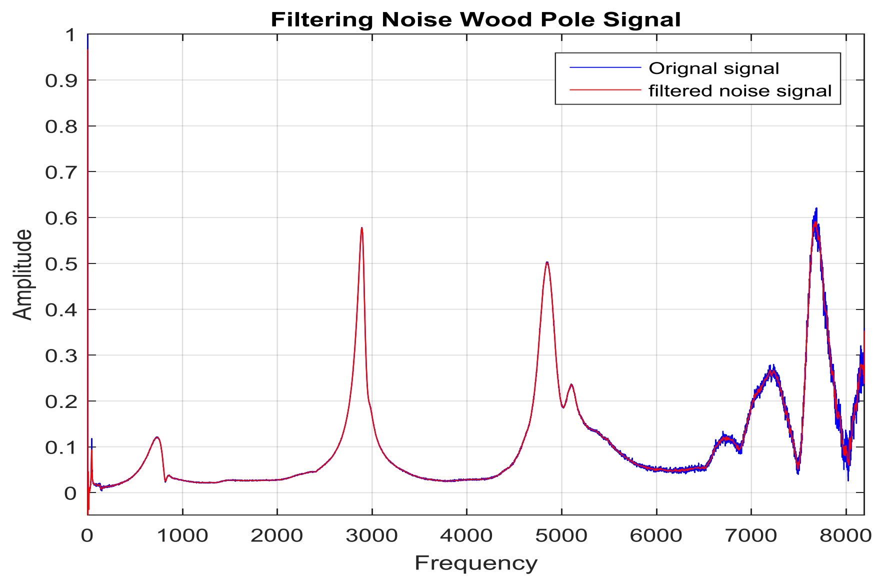


Figure 3.7 Filtering noise wooden pole signal

Figure 3.7 shows one of the curves from Figure 3.5; Comparing Figure 3.5 with Figure 3.7, it is obvious that it peaks after smoothing restrained signal distortion, yet keeps the curve shape. During testing, wooden pole displays low frequency feature, from 0 to 2500 Hz; therefore, low-pass filter let the low frequency signal pass through, restrain the high frequency signal. Majority of undesired signal has been eliminated after filtering. As discussed from Chapter 3.2, next step is to extract locations of natural frequencies at each peak. Shown as Figure 3.8, system picked 7 peaks from a single curve. It is important to know that even though FRF response curves have been filtered, the noise signal still exists. Therefore, function parameters of computer program should be modified from piece to piece to meet target peaks.

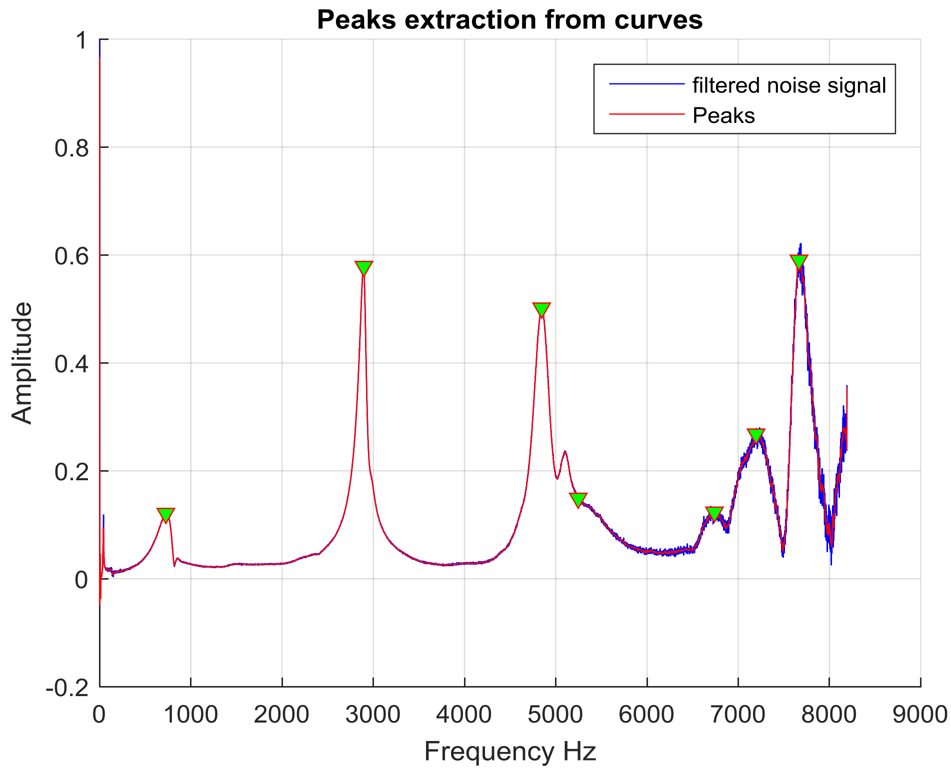


Figure 3.8 Extract Peaks from natural frequencies

Peak point 4th and 5th are not accurate peaks, they should be removed for further analysis. After selecting, a set of frequencies are extracted, shown at Table3.5. Each Curve may generate different numbers of peaks, each curve is required to check bad data extract, as in this case above, and peak 4 on Figure 3.4 should be deleted.

Table 3.5 Frequency extracted from curve

Curve-1	Normalized Amplitude	X-axis	Frequency(Hz)
Peak 1	0.1205	730	213.8411
Peak 2	0.5775	2892	847.1622
Peak 3	0.5015	4843	1418.674
Peak 4	0.2674	7199	2108.825
Peak 5	0.5910	7671	2247.089

3.2.2 Microphone Acoustic test

In the lab test, a single microphone was placed on a clamp about 30 cm away from a sample pole. On each pole, 20 points are marked from top to bottom for hammer hitting. Using 6 poles, 120 points in total, acoustic data has been collected Figure 3.5 shows sound wave collected by microphone in time domain.

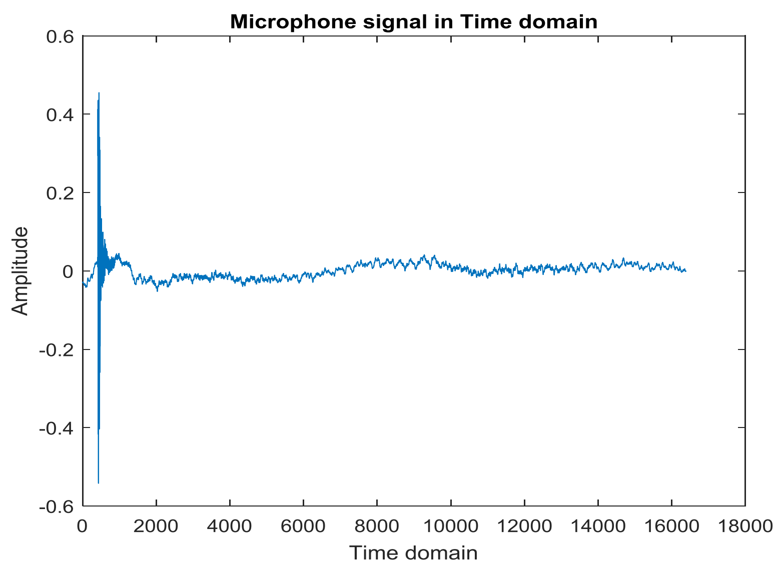


Figure 3.9 Original microphone signal

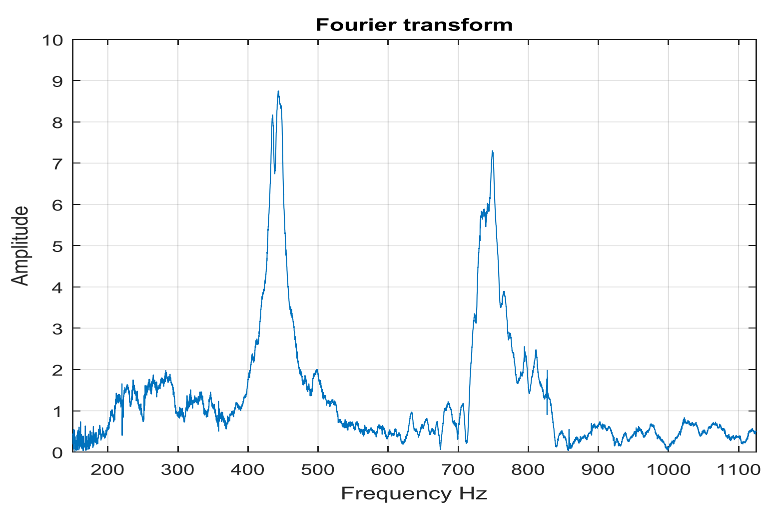


Figure 3.10 Signal after Fourier transform

In data processing, signal from time domain has the same volume of information. Normally, signal after Fourier transformation is straight forward for people to extract and analyze data from a certain frequency range. In wooden pole inspection, one would more care about the peaks' frequency responding location. From figure3.6, there are 4 major peaks in 0-2400 Hz. Thus, frequencies from those 4 peaks are useful for data training in neural network. Testers adapt new model decomposition algorithm based spectrum segmentation [11].

Model decomposition algorithm has two parts, one is Partial peak locating algorithm, another one is clustering algorithm. It uses concept of K-means algorithm, automatically gather clustering. $D(x, y)$ defines distance between peak and clustering.

$$D(B,C) = \frac{B_m C_m}{(B_x - C_x)^2} \tag{3.1}$$

Where, B_m is weight of current peak; C_m is weight of peak random peak; B_x is centroid of current peak; C_x is centroid of random peak.

Take N_B as total amount of peaks, sort peaks from biggest to lowest, take the biggest peak as initial cluster compare current peak centroid with both limits on left and right in the same cluster, it will get distance D(B,C). After several steps of iteration, the biggest value of peak merges current cluster of the particular peak as new cluster. Thus, new Modal spectral curve as:

$$X_k(w_s) = \frac{1}{\alpha_k^2} w_s^4 + 2\frac{\alpha_k^2 - w_{dk}^2}{\alpha_k^2} w_s^2 + \frac{(\alpha_k^2 + w_{dk}^2)^2}{\alpha_k^2} \tag{3.2}$$

Where, α_k is model parameter; w_s is frequency parameter; w_{dk} is random frequency parameter.

By applying iteration of distance of peak, would modify the range of cluster. Take model parameter from new cluster and peak to generate Modal Spectral Curve. In the particular case, eventually generates 4 clusters, shown as Figure 3.7. The values of new peaks are shown in Table 3.6.

Table 3.6 the values of new peaks

Peaks	Peak 1	Peak 2	Peak 3
Frequency(Hz)	293.8	599.1	766

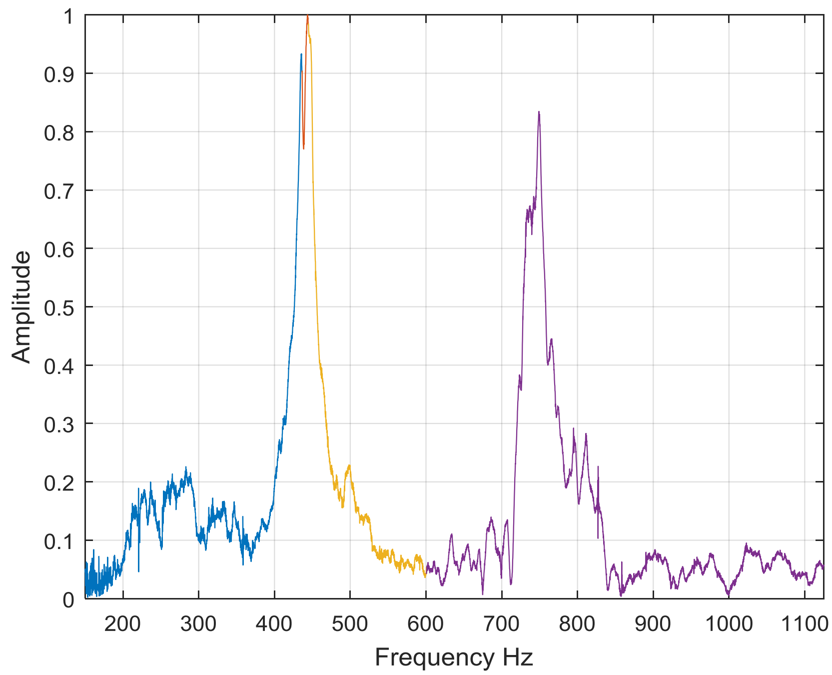


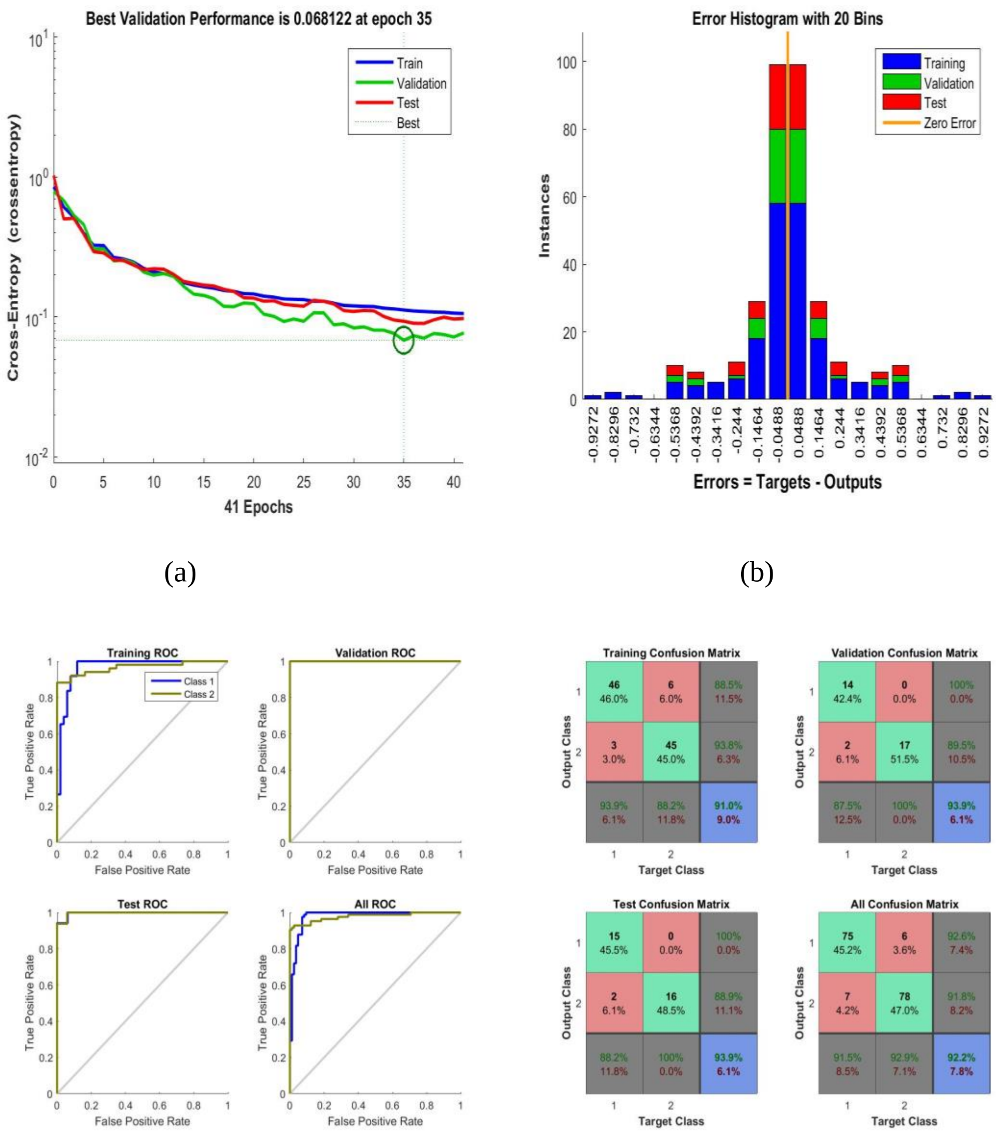
Figure 3.11 Peaks and Cluster

3.3 Neural Network Training

After extracting peaks from over 7 wooden pole samples, arrange peaks of same pole as column, match 7 poles as row. Take the peak data as input for neural network.

636.5	578.75	...	694.5	523	614	...	806	...		
1025.5	666.75	...	1038.75	605.5	815	...	1034.5	...		
1321.75	848.5	...	1144.25	778	975.25	...	1147	...		
1522.5	1097	...	1546	978	1093.75	...	1276	...		
1st hit of sensor 1	1st hit of sensor 2		1st hit of sensor 8	2nd hit of sensor 1			20th hit of sensor 8			

Figure 3.12 Table set of peaks



(a)

(b)

(c)

(d)

Figure 3.13 Network training results diagram

During neural network training, input data is separated into three groups. They are Training set, Validation set and Test set. Three set of data are distributing as default 60%, 20% and 20%. Figure 3.14 (a) shows at epochs 35, training data among validation data and test data reaches the smallest error. Meanwhile the training model reaches convergence after 41 iterations. The error stage is less than 10^-1. (b) shows errors at 0.0488 has most of values, that proves the training model is good enough to use. (c) and (d) are separated discussion for training, validation and test. From Matrix and ROC plots, three sets of data have over 90% accuracy; the training model is accurate enough for applying.

In order to verify the accuracy of the model, one selects 4 sets data from healthy wooden samples, and intentionally marks those data as unhealthy ones. The result shown as the value in matrix from Figure 3.15, only 12.5% to prove the model is accurate. On the other hand, the training model is a good training model identifying difference from healthy ones to unhealthy.

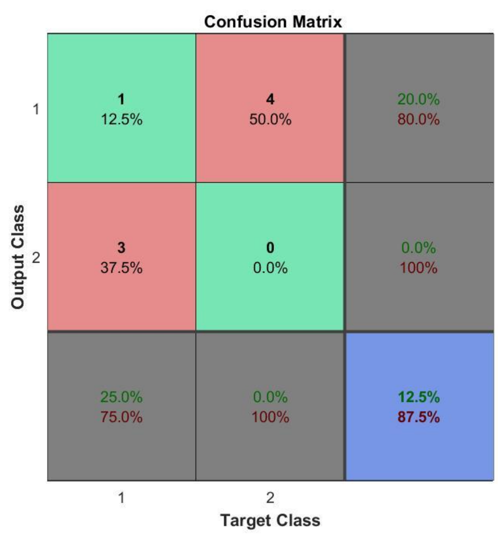


Figure 3.15 modeling testing result

3.4 Chapter Summary

Accelerometer vibration test uses different data processing algorithm from microphone acoustic test. This chapter introduced low pass filtering and Model decomposition algorithm, since different tools fit in different situations. Both of them collect frequency values of peaks from vibration signal and acoustic signal. We mark those frequency values as inputs to neural network as discussed in Chapter 2.

Chapter 4: Experiment Setup

4.1 Experiment Preparation

In the previous chapter, RNN and microphone test are mentioned. However, due to the model error is larger than 10^1 there would be inaccuracy and environment background noise. In this chapter, only Back Propagation Neural Network model and Accelerometer vibration test method will be covered for wooden pole inspection. At the beginning of experiment, hardware and its configuration will be mentioned.

For hardware, it has three major parts, data acquisition analyzer, and hammer for exciting vibration and accelerometer for receiving signal. In hardware introduction, hardware configuration and setup will be elaborated, including sensors installation height and angles, hammer exciting point, bad acquisition rejection condition.

4.2 Hardware selection and position placement

4.2.1 Data acquisition analyzer --Bobcat

In order to prepare for actual field testing, testers use Bobcat, portable model analyzer equipment, it can deal with various kinds of data extractions in time domain and in frequency domain, which meets the minimum field testing requirement.

Figure 4.1 Model Analyzer Bobcat

4.2.2 Force excitation transducer—PCB Hammer

There are four attached heads compare with hammer, testers conducted a small demonstration for each hammer head, from hard to soft (Hardness: Black> Red> Brown>Gray). The specific hammer configuration parameter is shown as Table 4.1.

Table 4.1 Configuration parameters for Force Transducer

Force Transducer	086D20
Sensitivity	1 mV/N
Measurement Range	±5000 N pk
Resonant Frequency	≥12 kHz
Constant Current Excitation	2 to 20 mA
Excitation Voltage	20 to 30 VDC
Hammer Mass	2.4 lb

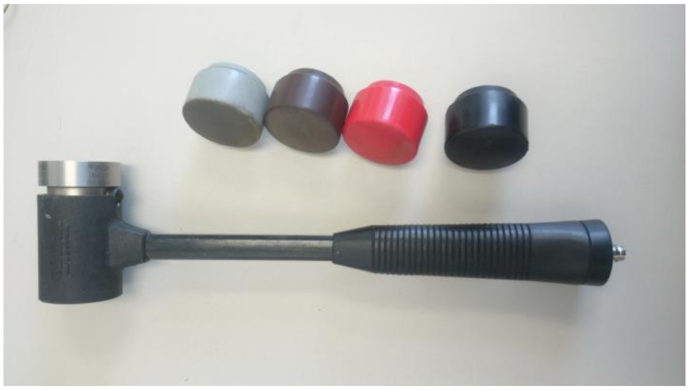


Figure 4.2 Force Transducer, according from left to right is from softest to hardest.

4.2.3 Vibration sensor—PCB accelerometer

Due to Bobcat design, it has four channels inputs, one channel is reserved for hammer, only three channels leave for accelerometers.

Figure 4.3 PCB accelerometer

Table 4.2 Accelerometer parameter Table

Accelerometer	PCB 352C66
Sensitivity	100 mV/g
Measurement Range	±50 g pk
Frequency Range	0.5 to 10000 Hz
Frequency Range	0.3 to 12000 Hz
Frequency Range	0.2 to 20000 Hz
Resonant Frequency	≥35 kHz
Phase Response	2 to 6000 Hz
Broadband Resolution	0.00016 g rms

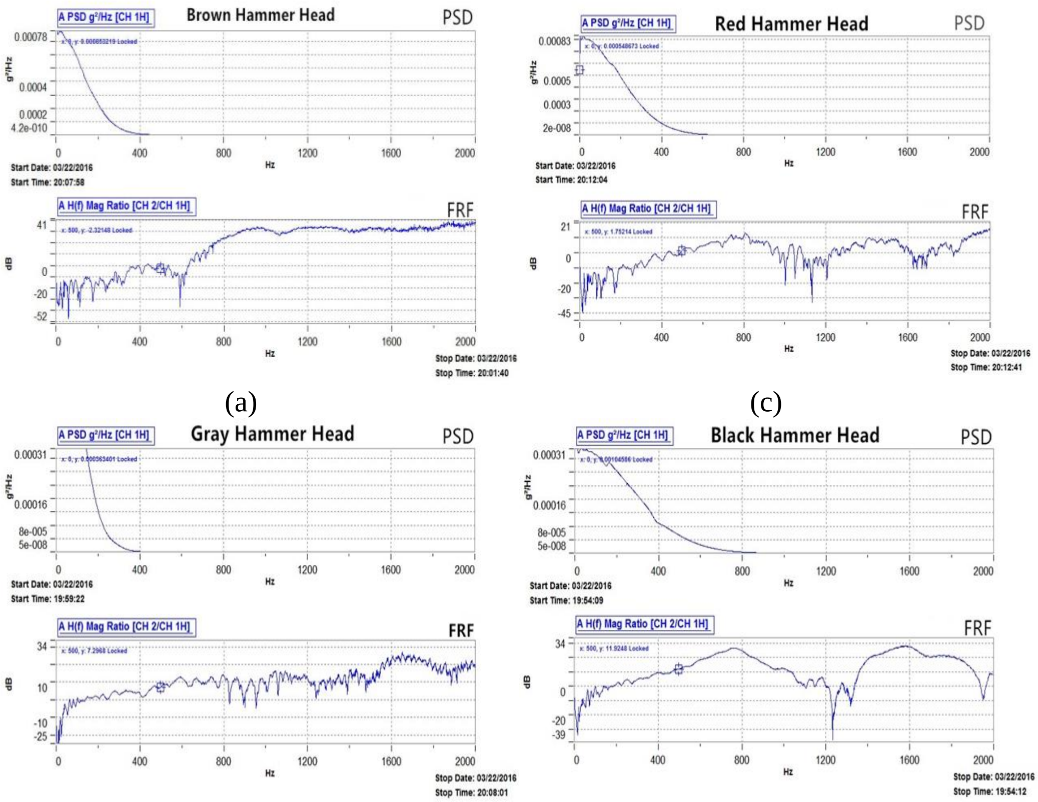
Table 4.3 Sensor placement on a single pole

Accelerometer	Sensor 2	Sensor 3	Sensor 4
Height （m）	0.2	1.5	1.5
Angle (°)	0	-120	120

Due to input channels limit, only three inputs can be used for accelerometers. The location of each accelerometer is referred on Table 4.3 and Figure 4.4. Three accelerometers are placed as 120° of a single layer circle. Sensor 2 is placed close to ground, because the damping ratio varies from measurement height, it enables sensor 2 to acquire from a separated angle from sensor 3 and 4.

4.3 Hammer Head Selection

There are various kinds of hammer that can be chosen. In order to have the right hammer head on the field test, one need to observe the excitation range over frequency axis and attenuation on the PSD figure. From figure3.4 below, from amplitude axis on FRF, they are in the same range, curves reflect properly by the excitation of different hammer head. In (a), frequency of the curve can last over to 400Hz; In (b), frequency of the curve can last to 400Hz, but the attenuation is higher than (a); In (c), frequency of the curve can last to 600Hz; In (d), frequency of the curve can last to 800Hz, which tells the effective sampling frequency of (d) is higher than any other. Figure3.4 (d) reflects more information of the curve.



(a) (c)

(b) (d)

Figure 4.4 Hammer excitation of different hammer heads with accelerometer response

Accelerometer response with hammer excitation range is a crucial to this project. Additionally, sensors' coherence of the excitation is another crucial standard in testing. As (a), (b), (c), (d) figures below, shows sensor coherence response with hammer excitation. Before the PSD curve reduce to zero, during that curve range, if the coherence curves of each sensor are closing to 1, it means accelerometer reflects more actual response to hammer excitation. From the figures below, (d) has a better trajectory match with less shake of signal.

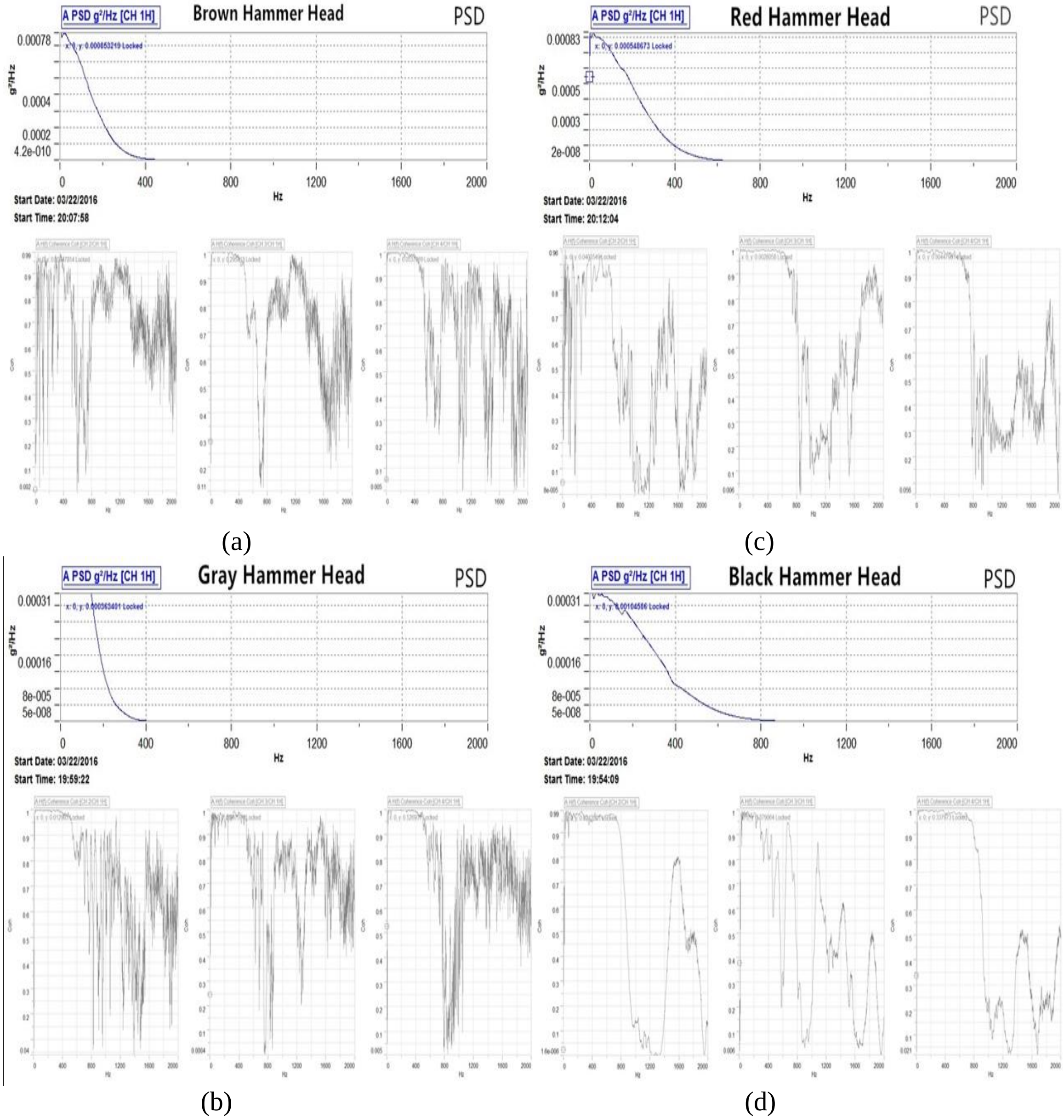


Figure 4.5 Hammer heads PSD with accelerometer coherence

Based on FRF and Coherence standard, the hardest hammer head (black hammer head) has the better effects in testing. The hammer excitation has a wider effective range till 800Hz, more information can be extracted from FRF, and more so, the accelerometers' coherence has a better match and less signal shake. Therefore, the black hammer head is the one to be applied in field.

4.4 Field testing

The field testing was conducted in Baltimore area, 65 poles are located in Ellicott City and 35 poles are located in Glen Burnie; These 100 poles aging between 2 years old to 80 years old. The entire testing scope has two phases, phase one, student collect data on the field, by using experiment equipment, such as Analyzer, hammer and accelerometer to collect wooden poles' oscillation data feedback. Phase two, analyzed collected FRF, time history data to conduct model analysis, then set featured values as input of neural network, simultaneously, set the target for neural network, the target data is provided by specialist of professional wooden pole inspection company. The target values include: "No damage", "Early stage of decay" and "Exposed Pocket". Those three conditions are labeled separately. By defining different conditions of wooden poles, the inspection company gives a reference standard.

Table 4.1 Wooden pole field inspection results

Number	BGE Code	OrigGL Circ	EffeGLCirc	Decay
1	97233	40	40	N
2	156008	39	39	N
4	174204	33.5	33.5	N
5	478991	38	38	N
6	174210	36	36	N
9	174212	24	24	N
10	174213	28	28	N
11	477958	33	33	N
12	214856	40	40	N
13	174215	34.5	33.5	Y
15	213685	41	41	N
16	247504	36	36	N

17	247505	32	32	N
18	170576	34	34	N
19	802502	36	36	N
20	802500	27.5	27.5	N
21	170573	32.5	32.5	N
22	170572	28	28	N
23	244984	36	36	N
24	170571	34	34	N
25	215848	34	34	N
26	565130	37	37	N
28	331121	40	40	N
29	336917	41.5	41.5	N
29	565137	37	37	N
30	272856	44.5	44.5	N
31	32334	44	44	N
32	451711	38	38	N
33	342222	36	36	N
34	811189	35	35	N
35	811190	41	41	N
36	502833	42	42	N
37	85898	35	35	N
38	85895	40	40	N
39	282602	38	38	N
40	195780	37	37	N
41	85899	45	45	N
42	57700	45	45	N
43	57701	39	39	N
44	57702	38	38	N
45	57703	41	41	N
46	57704	42	42	N
47	57707	45	45	N
48	265290	38	38	N
49	472924	35	35	N
50	813229	34	34	N
51	266060	36	36	N
52	451713	36	36	N
53	451712	39	39	N
54	540815	44	44	N
55	813227	43	43	N
56	334718	40	40	N
57	475840	38.5	38.5	N
58	813226	39	39	N
59	527883	38	38	N
60	483948	41	41	N
61	483949	39.5	39.5	N
62	483951	33.5	33.5	N

63	483947	35	35	N
64	230527	35	35	N
65	264167	35	35	N
66	333601	36	36	N
67	333602	32	32	N
68	333603	32	32	N
69	264166	40	40	N
70	316127	36	36	N
71	316126	38.5	38.5	N
72	316124	35	35	N
73	306233	36	36	N
74	175150	38	38	N
75	174084	28	28	N
76	812134	27.5	27.5	N
77	812434	28.5	27.5	Y
78	172023	36	36	N
79	316070	30	30	N
80	316092	36	36	N
81	373476	33.5	28	Y
82	316096	34.5	34.5	N
83	316122	31	28.09	Y
84	282477	32	32	N
85	129203	32	32	N
86	316105	32	32	N
87	316108	34	34	N
88	373322	36	36	N
89	316110	33.5	33.5	N
90	316111	40	40	N
91	316113	33.5	33.5	N
92	248179	39	39	N
93	373323	36	36	N
94	248182	33	33	N
95	373474	32	32	N
96	373475	33	33	N
97	353147	34	34	N
98	361197	32	32	N
99	362879	34.5	34.5	N
100	560325	39	39	N
101	265173	31	26.88	Y

Table 4.1 is a portion of inspection report provided by Inspection Company: 98 poles in total have been inspected; 93 poles are healthy and 5 poles are pocketed. From the report, visual inspection method is used to judge surface decaying

condition. If the surface of a pole shows beginning signs of corruption, then it will be labeled as early stage of decay, this means surface decay might have potential chance of advancing towards interior, to eventually generate a pocket. However, by only judging surface decay~~,~~ it is difficult to determine the pole is with certainty being unhealthy. Thus, the effective remaining length is introduced in Table 4.1. By comparing Effective GL circle length with Original GL circle length, once those two values are equal, the specific pole is a healthy one, Once Effective GL circle length is less than Original GL circle length, that specific pole is an unhealthy pocketed pole.

Namely, the poles labeled as “Early stage of decay” could be separated as healthy or pocketed.

During field testing, the accelerometer is place in 21 locations, shown as Figure 4.5. Each layer has 6 locations despite the hammering points on each layer.

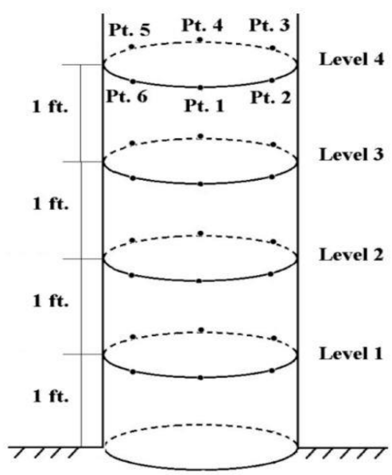


Figure 4.6 Measurement points assignment

Each hammer hitting point has three accelerometers data collections, shown as Figure 4.6. Namely, Sensor 2, Sensor 3, Sensor 4. Accelerometers are placed on the same level, 90 degrees' interval.

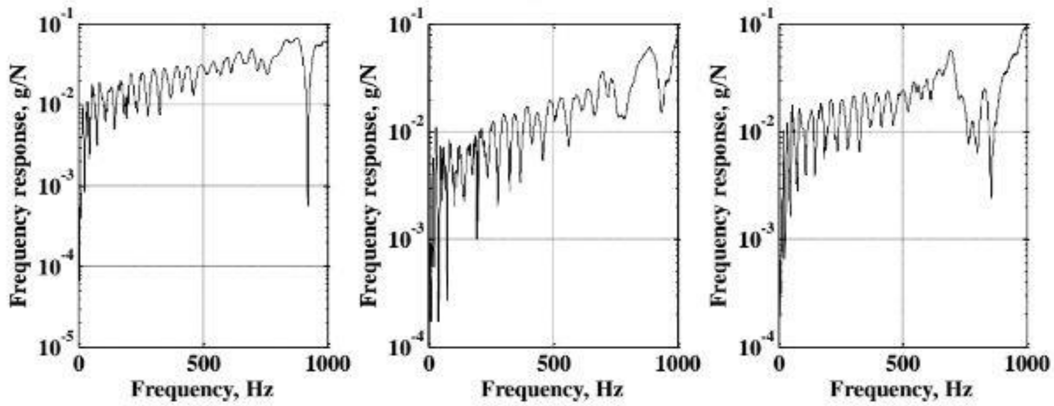


Figure 4.7 FRF feedbacks from accelerometers

Regarding of the data from Inspection Company, every three channels would be set as one group. Each pole has more than 20 groups, thus, for a single pole, there are 60 pieces of FRF data which can be collected. Eventually, there are 5 pocketed poles and 95 healthy poles included in field testing.

Due to FRF data doesn't have a very clear clue in picking peaks; however, Bobcat also catches Signal in Time domain, a single raw data is shown as Figure 4.8. Because of data in time domain is not readable, thus, time domain should covert to frequency domain via FFT transformation. Figure 4.8 is vibration response data in time domain for a single channel.

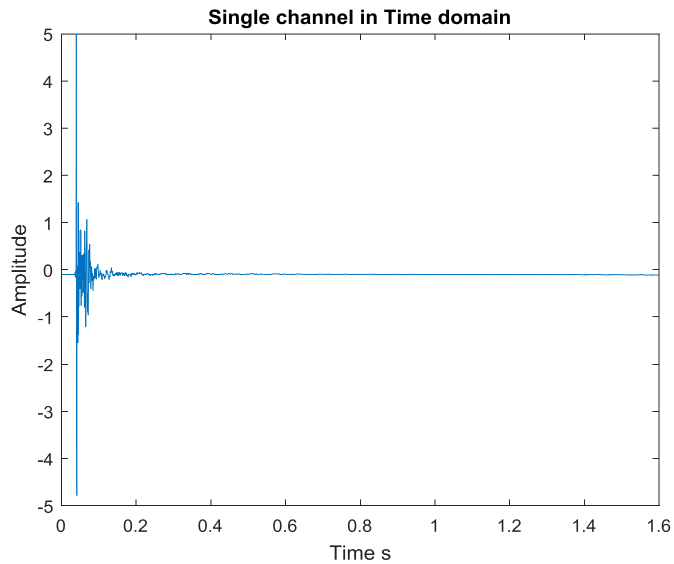


Figure 4.8 vibration responses in time domain

From Figure 4.9 a), effective range of Hammer can reach up to 1600 Hz. After classical FFT transformation, vibration response in frequency domain is shown as Figure 4.9 b). For a single point on one wooden pole, FFT response in frequency domain reaches up to 1250 Hz, the original frequency was up to 2500 Hz, due to FFT transformation has two sided and equal curve, thus, the effective FFT is half size of its range. Data from FFT transformation is shown as Figure 4.9.

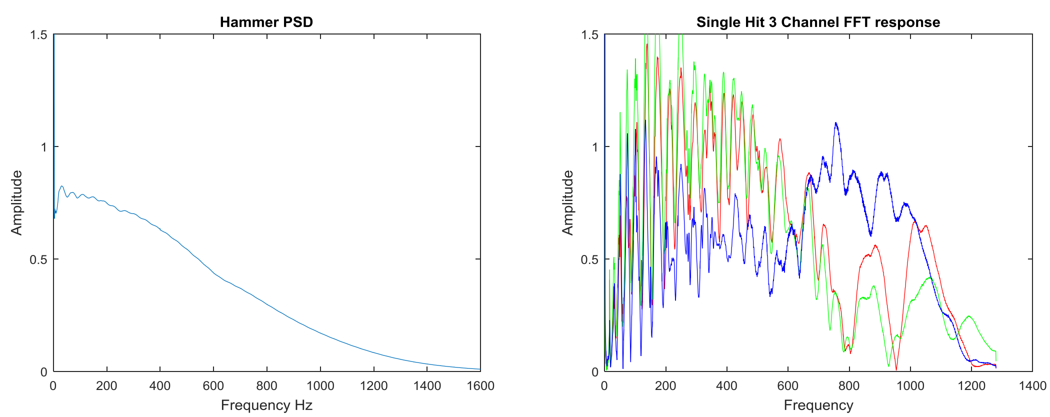


Figure 4.9 a) Hammer PSD b) 3 channels FFT converted vibration data

Cut the first several hundreds of noise FFT data, remaining FFT response is shown as Figure 4.10, in this Figure, low frequency response signal contains noise and background disturbance, thus, frequency from 0 to 500 Hz is cut.

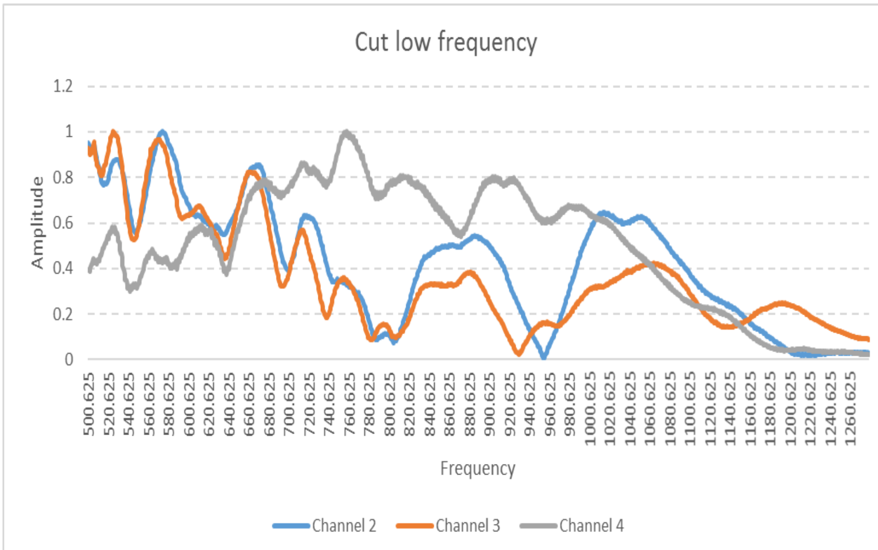


Figure 4.10 Refined vibration data in frequency domain for 3 channels

By collecting vibration data as Figure 4.10, group data in order of layers and categorized in different poles, then refer to the label set by BGE technician, then it is ready to put into neural network.

Neural network training

In previous chapter, the Back-Propagation and Recurrent Neural Network are introduced. The Recurrent Neural Network receives great training results in lab experiment. Yet for the data from field testing, the error of Cross-Entropy at the validation of model performance is larger than 10^1. In the scope of 100 poles, more than 10 poles are about to designated in the wrong group. Thus, Recurrent Neural

Network won't be covered in the following steps. Compare data sets from healthy and pocketed poles are shown in Table 4.2.

In Neural Network, higher model accuracy doesn't mean the model is more suitable for data validation. So, after building the model, additional performance tests should be conducted to validate the accuracy of neural network training. Once the additional performance tests remain the same or more accurate validation percentage, the model is a great neural network. Regarding of Table 4.2, a group from 95 healthy poles combined with 5 unhealthy poles has the highest model accuracy, however, in the additional performance test; the accuracy is less than 20% that is a bad neural network. The reason for this situation is because the number of healthy poles and unhealthy poles do not have a good peroration, data for training is 60%, data for Validation is 20% and data for Testing is 20%, the unhealthy poles would not be sufficient for validation or testing. In all, a proper percentage of data separation matters for neural network.

Table 4.2 Poles Percentage during Neural Network Training

Healthy and Unhealthy Poles Percentage				
Healthy poles	20	40	60	93
Unhealthy Poles	5	5	5	5
Model Accuracy	76%	80%	85%	100%

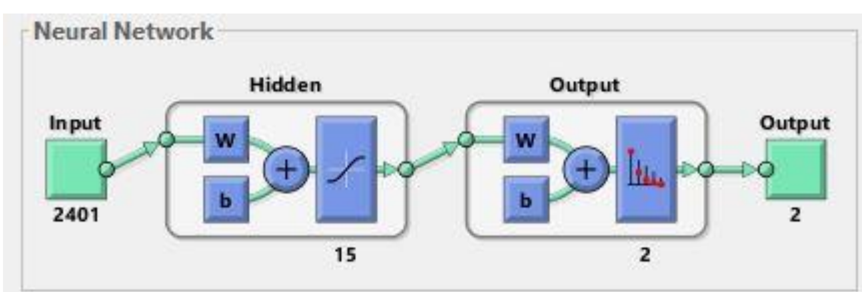
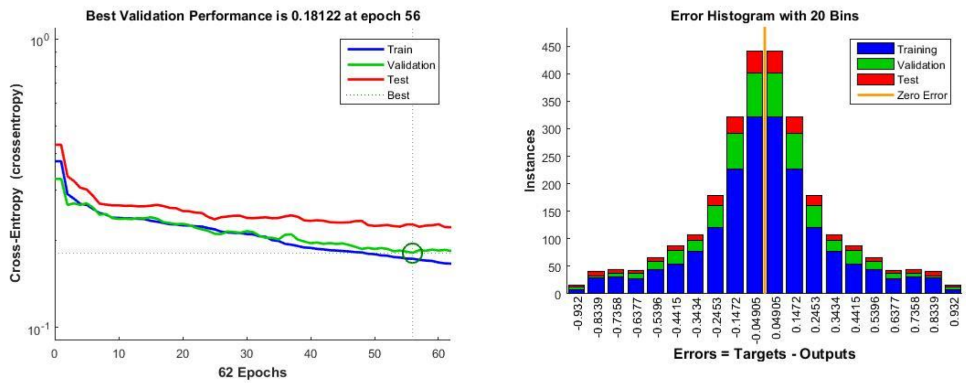


Figure 4.11 Back-Propagation Neural Network set up.



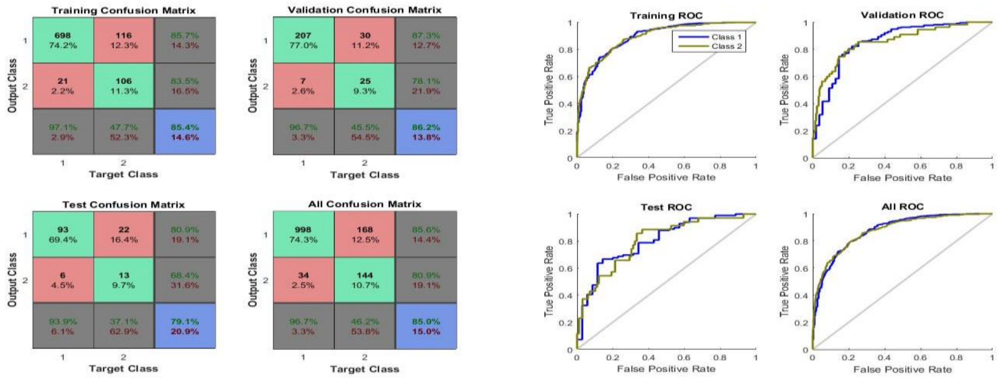
(a) (b)

Figure 4.12 Neural Network Model (a) Validation Performance and (b) Error Histogram

Shown as Figure 4.12a, it indicates the neural network model reaches convergence at the 56th epoch with no tendency of over-training. Difference between output value and target value is less than 10^0. While, the Error Histogram through another way proves the largest error is at 0.04905. There is a small gap between target and outputs, which can be neglected.

Confusion Matrix and ROC tells more about status and data in validation group, training group and testing group." All confusion Matrix" and "All ROC" are the ultimate result combined with those three groups. From Figure 4.12a shows, 85% of FRF data are accurately trained.

With additional data to validate this neural network model is accurate to reflect the poles inspection. Student extracted the FRF data from previously unused poles. Accuracies are higher than 80%, thus, this Backpropagation Neural Network model can be applied in the field wooden poles inspection.



(a) (b)

Figure 4.13 Neural Network Model (a) Confusion Matrix and (b) ROC Rate

After the model is validated, then get vibration response data from other un-used group to verify whether it's functioning. Randomly pick data from Pole 94 and input to neural network, the result shows 95% accuracy. Shown as Figure 4.14.

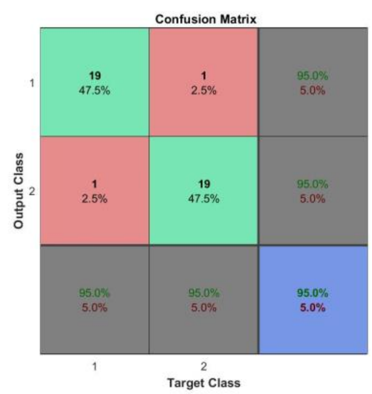


Figure 4.14 Neural network model further validations

Judging from results, the BPNN network can be applied for further data verification. Only by recording this BPNN model configuration parameter and extract the weights that have already been trained of this model, it can be applied to BGE technician for their daily inspection reference.

Chapter 5: Conclusion

In this work, an accurate and effective vibration-based wooden pole inspection method with neural network is developed, which would not cause any extra damage to wooden poles during inspection. Two vibration-measuring approaches, which are using microphone and accelerometers, are used to obtain data for neural network analysis in the lab testing, and results show that the current method can accurately and effectively identify healthy and unhealthy wood samples. Due to the complex environment background noises, data from microphone are no longer effective for neural network analysis and only data from accelerometers are obtained and analyzed using neural network. In data transformation, two peak extracting methods are applied. For data collected by accelerometer, Savitzky-Golay filtering and thread holding is applied; for data collected by Microphone, Model decomposition algorithm is adapted. Both of them have limitation in patch processing, all peaks should be confirmed manually. During field testing, microphone is affected by high volume background noise, thus, microphone couldn't be applied. On the other hand, two neural network models are compared each other, RNN has way much larger error state value, only BPNN is adapted in handling field testing. Final results show that the current vibration-based wooden pole inspection method with neural network is accurate and effective. With 85% of overall confusion matrix accuracy to be the highest accuracy value achieved and this model which would be applied for practical wooden pole inspection in the future.

Bibliography

[1] Rahman, A., and Chattopadhyay, G., 2007, "Soil Factors Behind Inground Decay of Timber Poles: Testing and Interpretation of Results," IEEE Transactions on Power Delivery, 22(3), pp. 1897-1903.

[2] Tomikawa, Y., Iwase, Y., Arita, K., and Yamada, H., 1986, "Nondestructive Inspection of a Wooden Pole Using Ultrasonic Computed Tomography," IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 33(4), pp. 354-358.

[3] Wyckhuyse, A., and Maldague, X., 2001, "A Study of Wood Inspection by Infrared Thermography, Part I: Wood Pole Inspection by Infrared Thermography," Journal of Research in Nondestructive Evaluation, 13(1), pp. 1-12.

[4] Wyckhuyse A, Maldague X., 2001, " A Study of Wood Inspection by Infrared Thermography, Part II: Thermography for Wood Defects Detection," Journal of Research in Nondestructive Evaluation, 13(1), pp. 13-22.

[5] Schafer R W, 2011, "On the Frequency-domain Properties of Savitzky-Golay Filters," Proceeding of the 2011 DSP/SPE Workshop, 2011, pp. 54-59.

[6] McGraw C M, Shroff H, Khalil G, et al., 2003, "The Phosphorescence Microphone: a Device for Testing Oxygen Sensors and Films," Review of Scientific Instruments, 74(12), pp. 5260-5266.

[7] Goh, A., 1995, "Back-Propagation Neural Networks for Modeling Complex Systems," Artificial Intelligence in Engineering, 9(3), pp. 143-151.

[8] Chen, F.-C., 1990, "Back-Propagation Neural Networks for Nonlinear Self-Tuning Adaptive Control," Control Systems Magazine, IEEE, 10(3), pp. 44-48.

[9] Abeles, M., 1991, Corticonics: Neural Circuits of the Cerebral Cortex, Cambridge University Press, New York.

[10] Jaeger, H., 2002, Tutorial on Training Recurrent Neural Networks, Covering Bppt, Rtrl, Ekf and the" Echo State Network" Approach, GMD-Forschungszentrum Informationstechnik.

[11] Qingwei, Y., Tongqing, W., and Junyong, Y., 2008, "A New Modal Decomposition Algorithm Based on Pso and Spectrum Segmentation," the 7th World Congress on Intelligent Control and Automation, pp. 1921-1926.