

# Classification of Various Wind Turbine Blade Faults through Vibration Signals Using Hyperpipes and Voting Feature Intervals Algorithm

Joshuva. A. \*, Sugumaran. V.

*School of Mechanical and Building Sciences (SMBS), VIT University, Chennai Campus, Vandalur-Kelambakkam Road, Chennai 600127, India.*

---

## Abstract

Wind energy has turned into a massive contender of traditional fossil fuel energy. Due to environmental conditions and over-time run conditions, wind turbine blades are prompt to different vibrations which cause damage to the blades. This paper presents an algorithmic classification of various blade fault conditions like blade bend, blade cracks, blade erosion, hub-blade loose connection and pitch angle twist using vibration signals. Initially histogram features were extracted from the vibration data and classified using machine learning algorithms like hyperpipes (HP) and voting feature intervals (VFI) algorithm. The performance of these algorithms were compared with respect to classification accuracy and better algorithm was suggested for fault prediction on wind turbine blades.

*Keywords:* Condition monitoring; fault diagnosis; voting feature interval algorithm; hyperpipes; histogram features; vibration signals.

(Submitted on January 1, 2017; Revised on March 28, 2017; Accepted on April 12, 2017)

© 2017 Totem Publisher, Inc. All rights reserved.

---

## 1. Introduction

Due to ecological imbalance, a tendency towards the feasible energy and renewable sources, for example, wind energy have risen worldwide. Wind power generation reduces the global warming issues because of its less ecological impact. Wind energy is one of the economical renewable energy sources and a possible alternative option for regular energy sources [1].

Wind power mounted capacity has a high development rate in all parts of the world. The main objective of the wind power plants is to increase the energy extraction from the wind. The power yield of the wind plants regularly changes with the influence of the winds. Moreover, the power production might also get affected by failure in wind turbine blades as they are the key segment to convert the kinetic energy to mechanical rotations [2]. This blade failure can also cause a huge catastrophic damage to the surroundings and may also damage the entire turbine framework. Hence, to perform condition monitoring of wind turbine blades while the turbine in working, machine learning approach is used. Machine learning can classify the particular failure, which occurred in blade at that instance with features extracted from vibration signals. [3].

Many studies and research work are reported in condition monitoring of wind turbine blades. To mention a few, Wang et al., [4] carried out a work on damage detection method for wind turbine blades based on dynamic analysis and mode shape difference curvature information using finite element methods (FEM). A study on wavelet transform based stress and time history editing of horizontal axis wind turbine blades was carried out by Pratumnopharat et al., [5]. With wavelet transform, this method extracts fatigue damage parts from the stress-time history and generates the edited stress-time history with the shorter time length. This study uses time correlated fatigue damage (TCFD), mexican hat wavelet (Mexh), meyer wavelet (Meyr), daubechies 30th order (DB30), morlet wavelet (Morl), discrete meyer wavelet (Dmey) for the classification of crack on the blade. Here, blade crack analysis was carried out.

A validation of an integral sliding mode control for optimal control of a three blade variable speed variable pitch wind turbine was done by Saravanakumar and Jena [6]. This study mainly focussed on the control of variable speed, variable pitch wind turbine for maximization of extracting power at below rated wind speed and regulation of extracting power when operating at above rated wind speed. Bitkina et al., [7] carried out an experimental and theoretical analysis of the stress strain

---

\* Corresponding author. Tel.: +91-9962648528; fax: NA.  
E-mail address: [joshuva1991@gmail.com](mailto:joshuva1991@gmail.com)

state of anisotropic multilayer composite panels for wind turbine blade. The stress-strain state of anisotropic multilayer composite panels (plates) was analyzed with temperature loadings using FORTRAN.

A design and kinetic analysis of wind turbine blade-hub-tower coupled system was carried out by Wenyi Liu [8]. The design was simulated for the 1.5 MW wind turbine and the kinetic analysis was carried out. Author focused on the blade and hub problem. Aero-structural design and optimization of a small wind turbine blade study was carried out by Pourrajabian et al., [9]. The study developed a methodology for aero-structural design, including consideration of the starting of a small wind turbine blade. This study carried out for both structural analysis and stress analysis on the blade by optimization.

Numerical investigation of the aerodynamic performance of a novel vertical axis wind turbine with adaptive blades was studied by Wang et al., [10]. This study made a novel Darrieus vertical axis wind turbine design whose blade can be deformed automatically into a desired geometry and they achieved a better aerodynamic performance. In this study performance was analyzed for vertical axis wind turbine (Darrieus) blade. Rezaei et al., [11] carried out a study on modal based damage identification for the nonlinear model of modern wind turbine blade. This study considered geometric nonlinearity due to large structural deformation of the modern wind turbine blade using a finite element model.

Numerous works were carried out using simulation analysis of fault and design analysis of wind turbine blade; however, only a very few in the experimental analysis was carried out. The machine learning technique was considered for wind turbine blade fault diagnosis; however, the usage was limited in the literature. A very limited set of defects was considered for analysis. This is especially true in the case of fault diagnosis of wind turbine blade. Hence, there is a strong need to design a fault diagnosis system which can handle multiple faults in wind turbine blades using machine learning approaches. This study makes an attempt to find 5 (five) different blade fault conditions by applying machine learning approach with histogram features. Figure 1 shows the methodology of the work carried out. The contribution of the present study is as follows:

- This study considers five faults (blade crack, erosion, hub-blade loose connection, pitch angle twist and blade bend) for wind turbine blade fault diagnosis.
- Histogram feature extraction tool was used to extract the required features from the vibration signals.
- J48 decision tree algorithm was used for feature selection.

This problem was modelled as a multiclass classification problem and attempts to classify using machine learning classifiers like hyperpipes (HP) and voting feature intervals (VFI) algorithms.

The rest of the paper is organized as follows. In section 2, the experimental setup and experimental procedure are explained. Section 3 presents the feature extraction process using histogram. The feature selection using J48 decision tree algorithm is presented in section 4. In section 5, the classifiers like hyperpipes (HP) and voting feature intervals (VFI) algorithms are explained. The results obtained from the classifiers and the discussion about their performance are presented in section 6. Conclusions are presented in the final section (section 7).

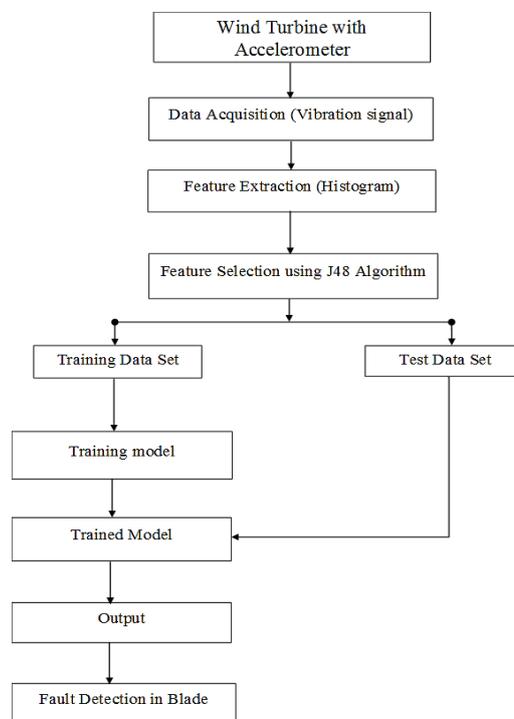


Figure 1. Flowchart for Blade Fault Diagnosis

## 2. Experimental Studies

The main aim of this study is to classify whether the blades are in good condition or in a defective state. If it is defective, then the objective is to identify the type of fault. The experimental setup and experimental procedure are described in the following subsections [12].

### 2.1. Experimental Setup

The experiment was carried out on a 50W, 12V variable speed wind turbine (MX-POWER, model: FP-50W-12V). The technical parameters of a wind turbine are given in Table 1. The wind turbine was mounted on a fixed steel stand in-front of the open circuit wind tunnel outlet. The wind tunnel speed ranges from 5m/s to 15 m/s and act as a wind source to start the wind turbine. The wind speed was varied continuously in order to simulate the environmental wind condition. Experimental setup is shown in Figure 2. Piezoelectric type accelerometer was used as transducer for acquiring vibration signals. It has high-frequency sensitivity for detecting faults. Hence accelerometers are widely used in condition monitoring.

In this case, a uniaxial accelerometer of 500g range, 100 mV/g sensitivity, and resonant frequency around 40 Hz was used. The piezoelectric accelerometer (DYTRAN 3055B1) was mounted on the nacelle near to the wind turbine hub to record the vibration signals using an adhesive mounting technique. It was connected to the DAQ system through a cable. The data acquisition system (DAQ) used was NI USB 4432 model. The card has five analog input channels with a sampling rate of 102.4 kilo samples per second with 24-bit resolution. The accelerometer is coupled to a signal conditioning unit which consists of an inbuilt charge amplifier and an analogue-to digital converter (ADC). From the ADC, the vibration signal was taken. These vibration signals were used to extract features through feature extraction technique. One end of the cable is plugged to the accelerometer and the other end to the AIO port of DAQ system. NI – LabVIEW was used to interface the transducer signal and the system (PC).

Table 1. Technical parameters of wind turbine

Model	FP-50W-12V
Rated Power	50 W
Rated Voltage	12 V
Rated Rotating Rate	850 rpm
Start-up Wind Speed	2.5 m/s
Cut-in Wind Speed	3.5 m/s
Cut-out Wind Speed	15 m/s
Security Wind Speed	40 m/s
Rated Wind Speed	12.5 m/s
Engine	Three-phase permanent magnet generator
Rotor Diameter	1050 mm
Blade Material	Carbon fiber reinforced plastics

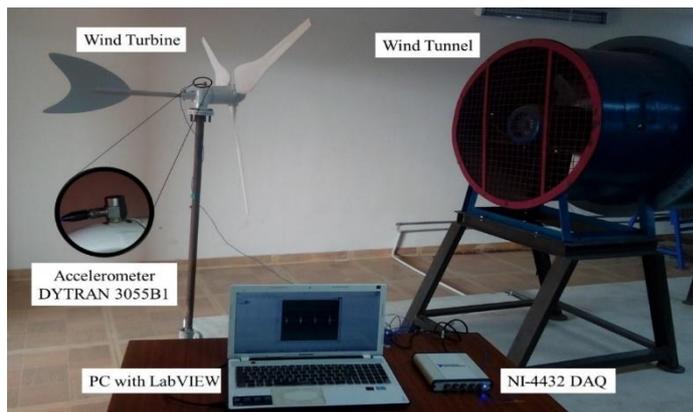


Figure 2. Experimental Setup

### 2.2. Experimental Procedure

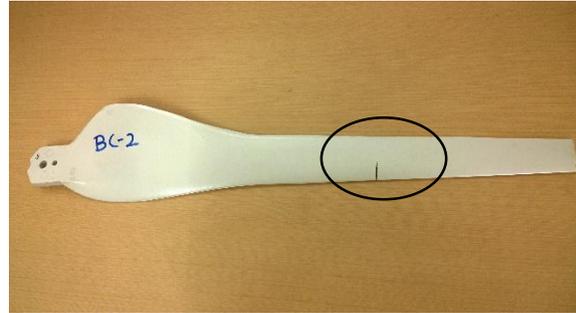
In the present study, three-blade variable horizontal axis wind turbine (HAWT) was used. Initially, the wind turbine was considered to be in good condition (free from defects, new setup) and the signals were recorded using an accelerometer. These

signals were recorded with the following specifications:

1. **Sample length:** The sample length was chosen long enough to ensure data consistency; and also the following points were considered. Statistical measures are more meaningful, when the number of samples is sufficiently large. On the other hand, as the number of samples increases the computation time increases. To strike a balance, sample length of 10000 was chosen.
2. **Sampling Frequency:** The sampling frequency should be at least twice the highest frequency contained in the signal as per Nyquist sampling theorem. By using this theorem sampling frequency was calculated as 12 kHz (12000Hz).
3. **Number of samples:** Minimum of 100 (hundred) samples were taken for each condition of the wind turbine blade and the vibration signals were stored in data files.



BG - Blade good condition



BC-2 – Blade crack



BPT – Blade pitch twist



BE – Blade erosion



BB – Blade bend (Front View)



Blade bend (Top View)

Figure 3. Various blade fault conditions

The vibration signals are acquired at 12000Hz with the sample size of 10000. The time taken for single data packet is roughly around 0.83s. The following faults were simulated one at a time while all other components remain in good condition and the corresponding vibration signals were acquired. Figure 3 shows the different blade fault conditions which are simulated on the blade.

- a) **Blade bend (BB):** This fault occurs due to the high-speed wind and complex forces caused by the wind. The blade was made to flap wise bend with  $10^0$  angles.

- b) **Blade crack (BC-2):** This occurs due to foreign object damage on blade while it is in operating condition. On blade, 15mm crack was made.
- c) **Blade erosion (BE):** This fault is due to the erosion of the top layer of the blade by the high-speed wind. The smooth surface of the blade was eroded using emery sheet (320Cw) to provide an erosion effect on the blade.
- d) **Hub-blade loose contact:** This fault generally occurs on a wind turbine blade due to an excessive runtime or usage time. The bolt connecting the hub and blade was made loose to obtain this fault.
- e) **Blade pitch angle twist (PAT):** This fault occurs due to the stress on the blade caused by high-speed wind. This makes the pitch get twisted, creating a heavy vibration to the framework. To attain this fault, blade pitch was twisted about 120 with respect to the normal blade condition.

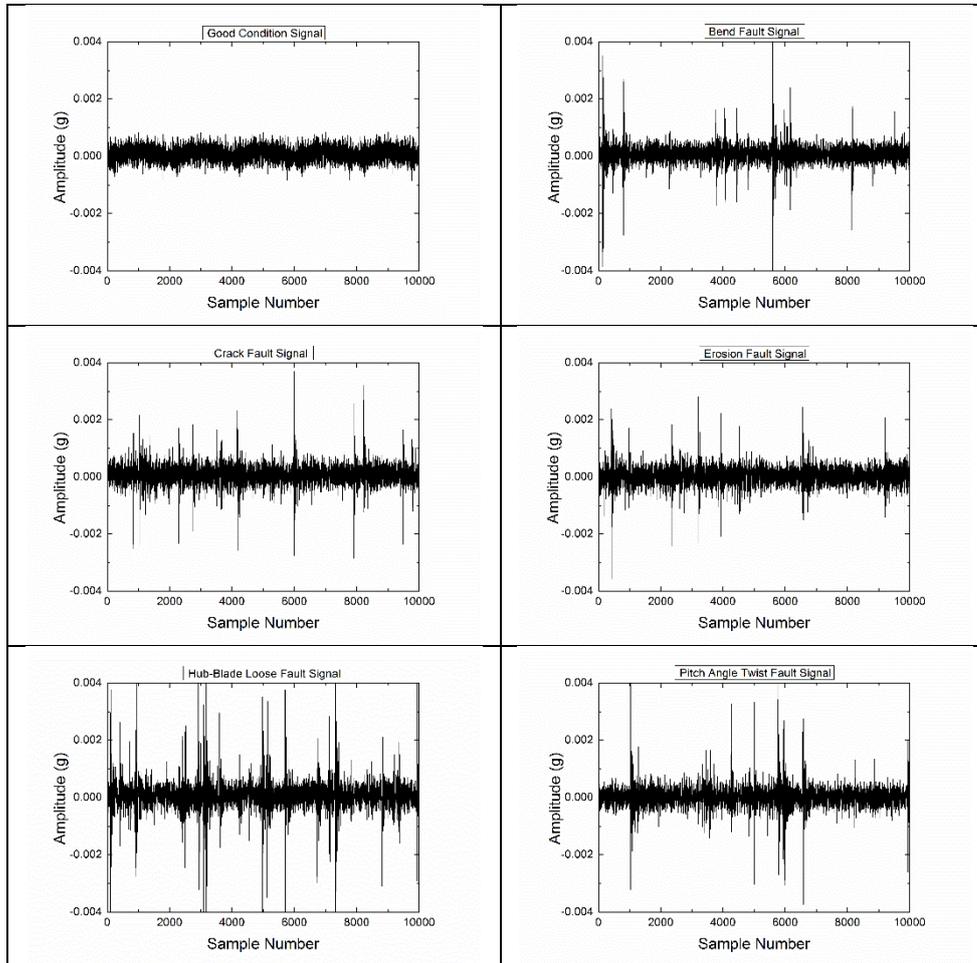


Figure 4. Vibration signal plot

Figure 4 shows the vibration signals which were taken from different conditions of the wind turbine blade like good condition blade, blade bend, blade erosion, hub-blade loose connection, blade crack and pitch angle twist respectively.

### 3. Feature Extraction

The vibration signals were obtained for good and other faulty conditions of the blades. If the time domain sampled signals are given directly as inputs to a classifier, then the number of samples should be constant. The number of samples obtained are the function of rotation of the blade speed. Hence, it cannot be used directly as the input to the classifier. However, a few features must be extracted before the classification process. The histogram was used as a feature extracting tool in this study. From the noted vibration signals, the needed feature is taken and that features are denoted as histogram features. There are two main factors to be considered in selection of bins they are, bin range and bin width [13].

The bin range must be from lowest of minimum amplitude (-0.017988) to extreme of maximum amplitude (0.024833) of all the six classes (good, bend, crack, erosion, loose and PAT). The number of bins for the fault diagnosis of wind turbine

blade has been attained by carrying out a sequence of trials using J48 algorithm with different number of bins. Initially, the range of bin is separated into two equivalent portions. That is to say, the number of bins utilized is two. The two histogram features, to be specific,  $X_1$  and  $X_2$  are extracted and the relating classification accuracy is additionally acquired by using J48 algorithm. The approach and methodology of performing the same using J48 algorithm are clarified in Section 4. A set of related trails is done with various numbers of bins from 2, 3, 4, 5, ..., 100 and the corresponding results are shown in figure 5.

From figure 5, bin size 77 has been chosen since the classification accuracy of bin 77 was found to be 92%. A set of 77 starting from  $X_1, X_2 \dots X_{77}$  were extracted from the vibration signals and these are denoted as histogram features. The amplitude ranges from -0.017988 to 0.024833. For further study, rather than utilizing vibration signals directly, the histogram features extracted from vibration signals are utilized. The procedure of calculating applicable parameters of the signals that represent the data contained in the signal is called feature extraction. Histogram analysis of vibration signals yields distinctive parameters. All the extracted histogram features,  $X_1$  to  $X_{77}$  extracted from the vibration signals may not contain the needed information for classification. The applicable ones are selected using J48 algorithm.

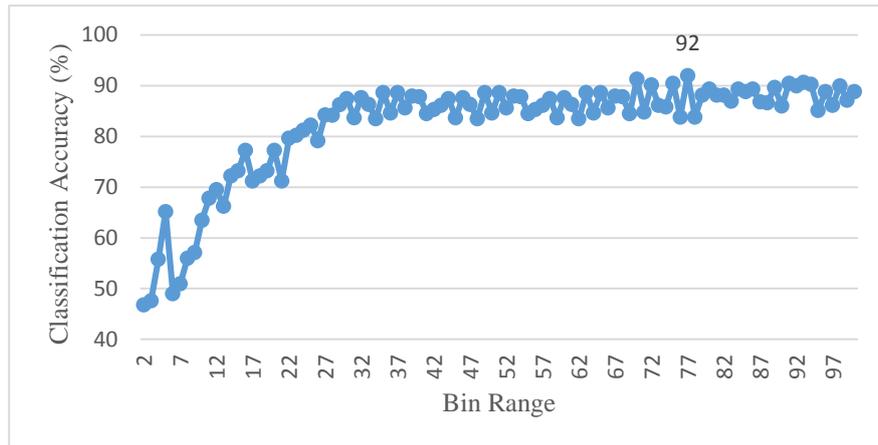


Figure 5. Bin Range vs Classification Accuracy

#### 4. Feature Selection

Data mining techniques are being increasingly used in many modern organizations to retrieve valuable knowledge structures from databases, including vibration data. An important knowledge structure that can result from data mining activities is the decision tree (DT) that is used for the classification of future events. Decision trees are typically built recursively, following a top-down approach. The acronym TDIDT, which stands for Top-Down Induction on Decision Trees, refers to this kind of algorithm. A standard tree induced with C5.0 (or possibly ID3 or C4.5) consists of a number of branches, one root, a number of nodes and a number of leaves. One branch is a chain of nodes from root to a leaf; and each node involves one attribute. The occurrence of an attribute in a tree provides the information about the importance of the associated attribute. J48 algorithm (a WEKA implementation of C4.5 algorithm) is a widely used one to construct decision trees [14].

The procedure of forming the decision tree and exploiting the same for vibration analysis is characterised by the following:

1. The set of histogram features extracted from wind turbine blade vibration studies forms the input to the algorithm; the output is the decision tree.
2. The decision tree has leaf nodes, which represent class labels, and other nodes associated with the classes (level of magnitude in this case) being analysed.
3. The branches of the tree represent each possible value of the parameter node from which they originate.
4. The decision tree can be used to express the structural information present in the data by starting at the root of the tree (top most node) and moving through a branch until a leaf node.
5. The level of contribution by each individual parameter is given by a histogram measure within the parenthesis in the decision tree (Figure 6). The first number in the parenthesis indicates the number of data points that can be classified using that parameter set. The parameters appearing in the nodes of decision tree are in descending order of importance.
6. At each decision node in the decision tree, one can select the most useful parameter for classification using appropriate estimation criteria.

The criterion used to identify the best parameter invokes the concept of entropy and information gain discussed in detail in the following subsections. Decision tree algorithm (C4.5) has two phases: building and pruning. The building phase is also called as the 'growing phase' [15].

#### 4.1. Building Phase

In the building phase, the training sample set with discrete-valued attributes is recursively partitioned until all the records in a partition have the same class. The tree has a single root node for the entire training set. Then for every partition, a new node is added to the decision tree. For a set of samples in a partition  $S$ , a test attribute  $X$  is selected for further partitioning the set into  $S_1, S_2, \dots, S_L$ . New nodes for  $S_1, S_2, \dots, S_L$  are created and these are added to the decision tree as children of the node for  $S$ . Also, the node for  $S$  is labelled with test  $X$ , and partitions  $S_1, S_2, \dots, S_L$  are then recursively partitioned. A partition in which all the records have identical class label is not partitioned further, and the leaf corresponding to it is labelled with the corresponding class. The construction of decision tree depends very much on how a test attribute  $X$  is selected. C4.5 uses entropy based information gain as the selection criteria. The entropy information gain is calculated in the following way.

Step-1: Calculate  $Info(S)$  to identify the class in the training set  $S$

$$Info(S) = -\sum_{i=1}^k \left\{ \left[ freq \left( C_i, \frac{S}{|S|} \right) \right] \log_2 \left[ freq \left( C_i, \frac{S}{|S|} \right) \right] \right\} \quad (1)$$

where  $|S|$  is the number of cases in the training set.  $C_i$  is a class,  $i = 1, 2, \dots, K$ .  $K$  is the number of classes and  $freq(C_i, S)$  is the number of cases included in  $C_i$ .

Step-2: Calculate the expected information value,  $Info_X(S)$  for test  $X$  to partition  $S$

$$Info_X(S) = -\sum_{i=1}^L \left[ \frac{|S_i|}{|S|} Info(S_i) \right] \quad (2)$$

where  $L$  is the number of outputs for test  $X$ ,  $S_i$  is a subset of  $S$  corresponding to the  $i^{th}$  output and is the number of cases of subset  $S_i$ .

Step-3: Calculate the information gain after partition according to test  $X$ :

$$Gain(X) = Info(S) - Info_X(S) \quad (3)$$

Step-4: Calculate the partition information value  $SplitInfo(X)$  acquired for  $S$  partitioned into  $L$  subsets

$$SplitInfo(X) = -\frac{1}{2} \left[ \sum_{i=1}^L \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|} + \left( 1 - \frac{|S_i|}{|S|} \right) \log_2 \left( 1 - \frac{|S_i|}{|S|} \right) \right] \quad (4)$$

Step-5: Calculate the gain ratio of  $Gain(X)$  over  $SplitInfo(X)$

$$GainRatio(X) = Gain(X) - SplitInfo(X) \quad (5)$$

The  $GainRatio(X)$  compensates for the weak point of  $Gain(X)$  which represents the quantity of information provided by  $X$  in the training set. Therefore, an attribute with the highest  $GainRatio(X)$  is taken as the root of the decision tree [16].

#### 4.2. Pruning Phase

A large decision tree constructed from a training set usually does not retain its accuracy over the whole sample space due to over-training or under-fitting. Therefore, a fully grown decision tree needs to be pruned by removing the less reliable branches to obtain better classification performance over the whole instance space even though it may have a higher error over the training set. The C4.5 algorithm uses an error-based post pruning strategy to deal with over-training problem [17]. For each classification node C4.5 calculates a kind of predicted error rate based on the total aggregate of misclassifications at that particular node. The error based pruning technique essentially reduces to the replacement of vast sub-trees in the classification structure by singleton nodes or simple branch collections if these actions contribute to a drop in the overall error rate of the root node.

#### 4.3. Discretisation of Continuous-Valued Attribute

It is important to know about how C4.5 solves the classification problem with continuous attributes because most of the signals in fault diagnosis field have continuous values. In fact, the discretisation process of continuous-valued attributes in C4.5 algorithm is a process to select the optimal threshold. For a continuous-valued attribute  $X$ , suppose it has  $m$  values in the training set and the values are sorted in ascending order, i.e.,  $\{a_1, a_2, \dots, a_m\}$  ( $a_1 \leq a_2 \leq \dots \leq a_m$ ). For a special value  $a_i$ , it partitions the samples into two groups  $(a_1, a_2, \dots, a_i)$  and  $(a_{i+1}, a_{i+2}, \dots, a_m)$ . One has  $X$  values up to  $a_i$ , the other has  $X$  values greater than  $a_i$  and  $a_i$  is an optional threshold for discretisation. Therefore, there exist  $m-1$  kinds of partitions or there are  $m-1$  thresholds available. For each of these partitions, compute the information gain (see Section 4.1) and choose the partition (given the  $j^{th}$  partition) that maximises the gain. Accordingly, the boundary value  $a_j$  in the optimal partition is selected as the optimal threshold. This dynamic discretisation method is executed for each candidate attribute in every process to select the best test attribute.

#### 4.4. Application of Decision Tree for Feature Selection

The algorithm has been applied to the problem under discussion for feature selection. Input to the algorithm is the set of features described in Section 4; the output is the decision tree, which is shown in Figure 6. It is clear there from that the top node is the best node for classification. The level of contribution is not same and all eleven features are not equally important. The level of contribution by each individual feature is given by a histogram measure within the parenthesis in the decision tree Figure 6. The first number in the parenthesis indicates the number of data points that can be classified using that feature set. The second number indicates the number of samples against this action. If the first number is very small compared to the total number of samples, then the corresponding features can be considered as outliers and hence ignored [18].

The other features appear in the nodes of decision tree in descending order of importance. It is to be stressed here that only features that contribute to the classification appear in the decision tree and others do not. Features that have less discriminating capability can be consciously discarded by deciding on the threshold. This concept is made use of in selecting good features. The algorithm identifies the good features for the purpose of classification from the given training dataset and thus reduces the domain knowledge required to select good features for pattern classification problem. A feature is ‘a good feature’, when its discriminating ability is high among the classes. It is characterised by the following:

- (a) The feature values do not vary much within a class.
- (b) It varies much among the classes.

The features which satisfy the above conditions will have more information gain while splitting and thus they appear in the order of importance in decision tree.

#### 4.5. Features Suggested by Decision Tree

The features that dominate generally represent the wind turbine blade condition descriptors. Referring to Figure 6, X33 is the most dominating feature followed by X34, X35, X36 and X32.

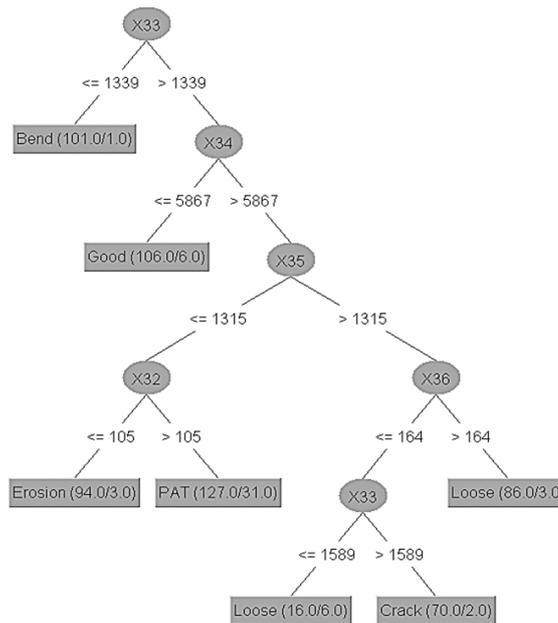


Figure 6. J48 Tree classification for feature selection

### 5. Feature Classification

After the feature selection, the fault classification was carried out using hyperpipes (HP) and voting feature intervals (VFI) algorithms. These both classifiers are called as miscellaneous Classifiers.

#### 5.1. Hyperpipes (HP)

HyperPipes is a very simple classification algorithm that shares the dual advantages of being both very fast in operation, and able to easily handle large quantities of attributes. The standard HyperPipes algorithm is unable to handle numeric class data,

a problem which is avoided by the discretization step in our process. The basic idea is to create a single “pipe” for each class of the dataset [19]. During training, the pipe for each class keeps track of which attribute values it has encountered thus far, but no counts of any kind are kept. Then, to test, each instance is classified according to the pipe that most contains its values, meaning the pipe that most matches the particular instance’s pattern of values. In practice, we have noticed that with full data sets this can often lead to all instances being attributed to one class, the class that has the most diverse values. For example, if a dataset contained a class that, anywhere in the set, saw every value at least once, every instance tested would fit into that pipe and would, therefore, be classified by that pipe’s class.

Testing on such a dataset would result in a very high recall rate, as every instance that actually was of the diverse class would be correctly identified, but would also yield an equally high false alarm rate, since every instance of a different class would be falsely classified [20]. Not keeping track of occurrence frequencies or totals has advantages and disadvantages. This technique is often considered “dumb” or “unsophisticated” due to the fact that a value that has been seen ten thousand times is of equal importance as one that has been seen only once. Also, there are no complicated formulas involving magical parameters that have been painstakingly calculated over years of research. On the other hand, the algorithm is extremely fast, has a minimal memory footprint, and is exceedingly simple.

5.2. Voting Feature Interval (VFI)

The VFI classification algorithm represents a concept description by a set of feature value intervals. The classification of a new instance is based on a voting among the classifications made by the value of each feature separately. It is a non-incremental classification algorithm; that is, all training examples are processed at once. From the training examples, the VFI algorithm constructs intervals for each feature. An interval is either a range or point interval. A range interval is defined for a set of consecutive values of a given feature whereas a point interval is defined for a single feature value [21]. For point intervals, only a single value is used to define that interval. For range intervals, on the other hand, it suffices to maintain only the lower bound for the range of values, since all range intervals on a feature dimension are linearly ordered. The lower bound of the range intervals obtained from the training instances are installed into an array called interval lower and the number of segments formed for each feature value is stored in the array No Intervals directly at the beginning of the VFI function so no training process is done. For each interval, a single value and the votes of each class in that interval are maintained. Thus, an interval may represent several classes by storing the vote for each class. The votes given to the classes for each interval for each feature values are stored in the interval votes array [22].

6. Results and Discussion

The vibration signals were noted for good condition and faulty blade conditions using DAQ. Totally 600 samples were collected; out of which 100 samples were from good condition blade. For different faults such as like blade bend, erosion, blade crack, hub-blade loose connection, pitch angle twist, 100 samples from every condition were noted. J48 decision tree algorithm was used to select the best contributing histogram features from bin size 77. From Figure 5 and 6, the selected features are given as the input to the classifier to determine the classification accuracy with respect to faults created on the wind turbine blade [23]. From Figure 7, hyperpipes (HP) gives the maximum classification accuracy of 87.67% when compared to voting feature intervals (VFI) algorithms (86.13%). The confusion matrix of the hyperpipes (HP) is shown in Table 2 and voting feature intervals (VFI) algorithms shown in Table 3. In confusion matrix, the diagonal element represents the correctly classified instance and the others are misclassified [24].

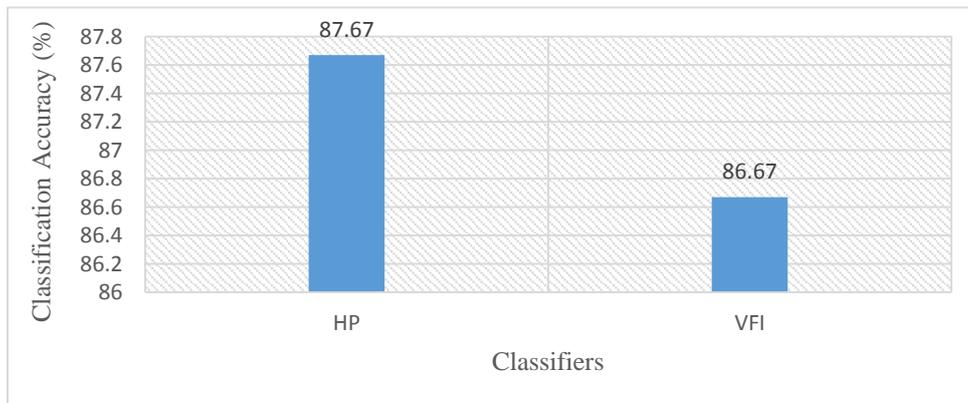


Figure 7. Overall classification accuracy of the classifiers

Table 2. Confusion matrix for hyperpipes (HP)

Blade conditions	Good	Bend	Crack	Erosion	Loose	PAT
Good	<b>97</b>	0	1	0	2	0
Bend	0	<b>100</b>	0	0	0	0
Crack	0	0	<b>96</b>	0	3	1
Erosion	0	7	4	<b>88</b>	0	1
Loose	3	1	12	0	<b>84</b>	0
PAT	0	1	36	2	0	<b>61</b>

Table 3. Confusion matrix for voting feature intervals (VFI)

Blade conditions	Good	Bend	Crack	Erosion	Loose	PAT
Good	<b>94</b>	0	1	0	5	0
Bend	0	<b>100</b>	0	0	0	0
Crack	0	0	<b>75</b>	2	11	12
Erosion	0	2	3	<b>94</b>	0	1
Loose	10	1	3	0	<b>84</b>	2
PAT	0	0	14	6	7	<b>73</b>

For HP, out of 600 samples, 526 samples are correctly classified (87.67%) and remaining 74 are misclassified (12.33%). For VFI, out of 600 samples, 520 samples are correctly classified (86.67%) and remaining 80 are misclassified (13.33%). In misclassification (HP), for pitch angle twist condition and in loose condition, 36 and 12 data sets show that it is categorized in a crack fault condition which is a serious condition since the algorithm classified the PAT and loose signals as crack fault. For VFI, in misclassification, for pitch angle twist condition, 14 datasets show that it is categorized in a crack fault condition and in crack condition, 12 datasets shows it has been misclassified as pitch angle twist fault. This misclassification can be reduced by training more samples to the algorithm. The time taken to build the HP model is about 0.02 seconds when compared to VFI model (0.08s); hence, HP model can use in real time for the fault detection on the wind turbine blade. The detailed classwise accuracy of HP model is shown in Table 4 and for VFI model, in Table 5.

Table 4. Classwise accuracy of hyperpipes (HP)

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC area
Good	0.970	0.006	0.970	0.970	0.970	0.991
Bend	1.000	0.000	1.000	1.000	1.000	1.000
Crack	0.960	0.106	0.644	0.960	0.771	0.914
Erosion	0.880	0.004	0.978	0.880	0.926	0.893
Loose	0.840	0.010	0.944	0.840	0.889	0.931
PAT	0.610	0.004	0.968	0.610	0.748	0.942

For HP model, the kappa statistics were found to be 0.852. It is used to measure the arrangement of likelihood with the true class. The mean absolute error was found to be 0.2333. It is a measure used to measure how close forecasts or prediction are with the ultimate result. The root mean square error was found to be 0.323. It is a quadratic scoring rule which processes the average size of the error. For VFI model, the kappa statistics were found to be 0.840, the mean absolute error was found to be 0.1613 and the root mean square error was found to be 0.253.

Table 5. Classwise accuracy of voting feature intervals (VFI)

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC area
Good	0.940	0.020	0.904	0.940	0.922	0.994
Bend	1.000	0.000	1.000	1.000	1.000	1.000
Crack	0.750	0.042	0.781	0.750	0.765	0.951
Erosion	0.940	0.016	0.922	0.940	0.931	0.996
Loose	0.840	0.046	0.785	0.840	0.812	0.937
PAT	0.730	0.030	0.830	0.730	0.777	0.951

The true positive (TP) rate should be close to 1 and the false positive (FP) rate should be close to 0 to propose the classifier is a better classifier for the problem classification [25]. TP and FP are very important in the classification process. In the hyperpipes (HP) algorithm, it shows that the TP near to 1 and FP close to 0, then one can predict that the classifier we build for the particular problem is very much effective for the condition monitoring of blade [26]. The classifier error chart is shown in Figure 8. Here the squared dots represent the misclassification and the 'x' denotes the correct classification.

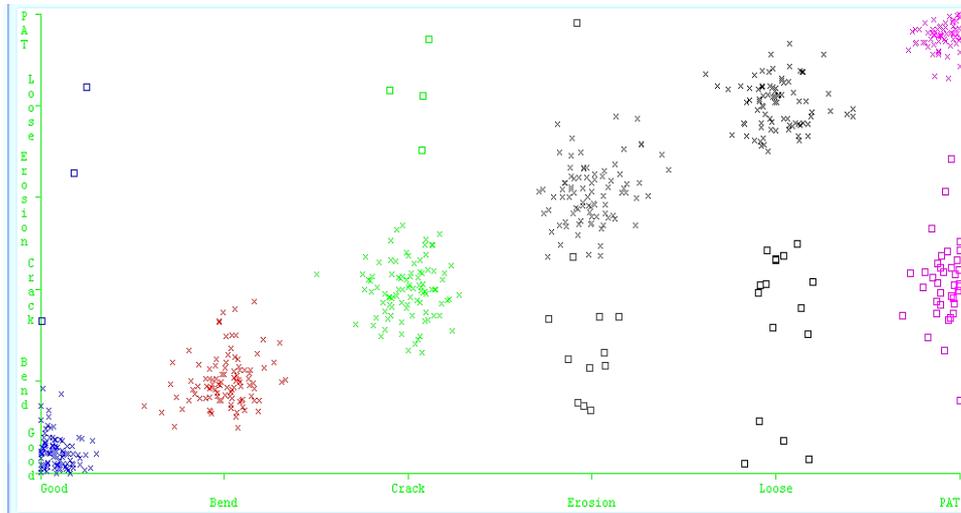


Figure 8. Classifier Errors for HP (Classification vs Misclassification)

## 7. Conclusion

The wind turbine is very important in the production of wind energy in day to day life. This paper presents an algorithmic based clarification of vibration signals for the assessment of wind turbine blade conditions. From the acquired vibration data, two models were developed using data modelling technique. Hyperpipes (HP) and voting feature intervals (VFI) algorithms was used to learn and classify the different conditions of the blade. These models are tested under 10-fold cross validation and maximum correctly classified instances were found to be 87.67% for hyperpipes (HP) algorithm. The error rate is relatively less and may be considered for the blade fault diagnosis. Hence, hyperpipes (HP) algorithm can be practically used for the condition monitoring of wind turbine blade to reduce the downtime and to provide more wind energy. The methodology and algorithm suggested in this paper can be potentially used for any kind of wind turbine blade to diagnose the blade fault with minimal modification.

## References

1. Ciang CC, Lee JR, Bang HJ. Structural health monitoring for a wind turbine system: a review of damage detection methods. *Measurement Science and Technology*. 2008 Oct 13;19(12):122001.
2. Tummala A, Velamati RK, Sinha DK, Indraja V, Krishna VH. A review on small scale wind turbines. *Renewable and Sustainable Energy Reviews*. 2016 Apr 30;56:1351-71.
3. Joshuva A, Sugumaran V. Fault diagnostic methods for wind turbine: A review. *Asian Research Publishing Network (ARPN) Journal of Engineering and Applied Sciences*. 2016 Apr;11(7):4654-68.
4. Wang Y, Liang M, Xiang J. Damage detection method for wind turbine blades based on dynamics analysis and mode shape difference curvature information. *Mechanical Systems and Signal Processing*. 2014 Oct 3;48(1):351-67.
5. Pratumnopharat P, Leung PS, Court RS. Wavelet transform-based stress-time history editing of horizontal axis wind turbine blades. *Renewable Energy*. 2014 Mar 31;63:558-75.
6. Saravanakumar R, Jena D. Validation of an integral sliding mode control for optimal control of a three blade variable speed variable pitch wind turbine. *International Journal of Electrical Power & Energy Systems*. 2015 Jul 31;69:421-9.
7. Bitkina O, Kang KW, Lee JH. Experimental and theoretical analysis of the stress-strain state of anisotropic multilayer composite panels for wind turbine blade. *Renewable Energy*. 2015 Jul 31;79:219-26.
8. Liu W. Design and kinetic analysis of wind turbine blade-hub-tower coupled system. *Renewable Energy*. 2016 Aug 31;94:547-57.
9. Pourrajabian A, Afshar PA, Ahmadizadeh M, Wood D. Aero-structural design and optimization of a small wind turbine blade. *Renewable Energy*. 2016 Mar 31;87:837-48.
10. Wang Y, Sun X, Dong X, Zhu B, Huang D, Zheng Z. Numerical investigation on aerodynamic performance of a novel vertical axis wind turbine with adaptive blades. *Energy Conversion and Management*. 2016 Jan 15;108:275-86.

11. Rezaei MM, Behzad M, Moradi H, Haddadpour H. Modal-based damage identification for the nonlinear model of modern wind turbine blade. *Renewable Energy*. 2016 Aug 31;94:391-409.
12. Joshuva A, Sugumaran V. A data driven approach for condition monitoring of wind turbine blade using vibration signals through best-first tree algorithm and functional trees algorithm: A comparative study. *ISA transactions*. 2017 Mar 31;67:160-72.
13. Joshuva A, Sugumaran V, Amarnath M. Selecting kernel function of Support Vector Machine for fault diagnosis of roller bearings using sound signals through histogram features. *International Journal of Applied Engineering Research*. 2015;10(68):482-7.
14. Mitchell TM. *Machine learning*. 1997. Burr Ridge, IL: McGraw Hill. 1997;45:37.
15. Witten IH, Frank E. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann; 2005.
16. Villacampa O. *Feature Selection and Classification Methods for Decision Making: A Comparative Analysis*. 2015.
17. Sakthivel NR, Sugumaran V, Babudevasenapati S. Vibration based fault diagnosis of monoblock centrifugal pump using decision tree. *Expert Systems with Applications*. 2010 Jun 30;37(6):4040-9.
18. Sugumaran V, Ramachandran KI. Fault diagnosis of roller bearing using fuzzy classifier and histogram features with focus on automatic rule learning. *Expert Systems with Applications*. 2011 May 31;38(5):4901-7.
19. Güvenir HA, Emeksiz N. An expert system for the differential diagnosis of erythematous diseases. *Expert Systems with Applications*. 2000 Jan 31;18(1):43-9.
20. Deeb ZA, Devine T, Geng Z. *Randomized Decimation HyperPipes*. Citeseer. 2010.
21. Demiröz G, Güvenir HA. Classification by voting feature intervals. In *European Conference on Machine Learning 1997 Apr 23 (pp. 85-92)*. Springer Berlin Heidelberg.
22. Ali F, Hayat M. Classification of membrane protein types using Voting Feature Interval in combination with Chou's Pseudo Amino Acid Composition. *Journal of theoretical biology*. 2015 Nov 7;384:78-83.
23. Joshuva A, Sugumaran V. Wind Turbine Blade Fault Diagnosis Using Vibration Signals through Decision Tree Algorithm. *Indian Journal of Science and Technology*. 2016 Dec 29;9(48).
24. Joshuva A, Sugumaran V, Amarnath M, Lee SK. Remaining Life-Time Assessment of Gear Box Using Regression Model. *Indian Journal of Science and Technology*. 2016 Dec 28;9(47).
25. Sugumaran V, Jain D, Amarnath M, Kumar H. Fault diagnosis of helical gearbox using Decision Tree through vibration signals. *International Journal of Performability Engineering*. 2013 Mar 1;9(2):221-34.
26. Elasha F, Teixeira Ja. Condition monitoring philosophy for tidal turbines. *International Journal of Performability Engineering*. 2014 Jul 1;10(5):521.

## Biographies

**A. Joshuva** completed his Bachelor of Engineering in Electronics and Communication Engineering from Anna University Affiliated College, Chennai, 2012. He completed his Masters of Business Administration in Human Resource Management from University of Madras, 2014. He pursued his Master of Technology in Mechatronics Engineering at VIT University, Chennai Campus, from 2013-2015. At present he is pursuing his doctoral degree in VIT University, Chennai Campus, Tamil Nadu. His research areas are Fault Diagnosis, Machine Learning, Data Mining, Pattern Recognition, Wind Turbine Condition Monitoring, Renewable Energy and Signal Processing.

**Dr. V. Sugumaran** received the B.E. degree in Mechanical Engineering from the Amrita Institute of Technology & Science, 1998 and the M. Tech in Production Engineering, from The National Institute of Engineering, 2003. Ph.D. degree in Fault Diagnosis, from Amrita School of Engineering, Amrita University, Coimbatore, Tamil Nadu, India, in 2008. From 2000 to 2009, he was an Associate Professor, with the Amrita School of Engineering, Coimbatore, India. From 2009 to 2011, he was working with SRM University, Chennai, Tamil Nadu. Since 2011, he has been an Associate Professor with the School of Mechanical and Building Sciences, VIT University, Chennai, Tamil Nadu, India. He is the author of one books, more than 150 international journal publications. He has also filed sixteen patents. His research interests include Condition Monitoring & Fault Diagnosis, Machine learning / Data mining in manufacturing and Mechanical Engineering. He presented papers in 43 International / National Conferences. He is also acting as a Reviewer for many international journals and editor for four international journals.