 SDHM, vol.11, no.2, pp.191-208, 2017

Classifying Machine Learning Features Extracted from Vibration Signal with Logistic Model Tree to Monitor Automobile Tyre Pressure

P. S. Anoop[1] and V. Sugumaran[2]

Abstract: Tyre pressure monitoring system (TPMS) is compulsory in most countries like the United States and European Union. The existing systems depend on pressure sensors strapped on the tyre or on wheel speed sensor data. A difference in wheel speed would trigger an alarm based on the algorithm implemented. In this paper, machine learning approach is proposed as a new method to monitor tyre pressure by extracting the vertical vibrations from a wheel hub of a moving vehicle using an accelerometer. The obtained signals will be used to compute through statistical features and histogram features for the feature extraction process. The LMT (Logistic Model Tree) was used as the classifier and attained a classification accuracy of 92.5% with 10-fold cross validation for statistical features and 90.5% with 10-fold cross validation for histogram features. The proposed model can be used for monitoring the automobile tyre pressure successfully.

Keywords: Machine learning, Vibration, accelerometer, Statistical Features, Histogram Features, Logistic model tree (LMT), Tyre pressure monitoring system.

1 Introduction

Tyres are the most important part of an automobile. A study led by the UK tire industry gathering censured that between the time of 2003 to 2005 around 50% of light vehicle crashes were because of flawed tires [Paine, Griffiths and Magedara (2007)]. The tyre pressure monitoring systems (TPMS) are intelligent electronic subsystems fitted on automobiles to monitor the tyre pressure on a real-time basis. According to the method of data acquisition, they are mainly classified into two categories viz. direct and indirect systems. TPMS are capable of giving real-time tyre-pressure information to the driver, using a simple low-pressure warning light or an advanced liquid crystal display [Velupillai and Güvenç (2007)]. Direct tyre pressure monitoring systems depend on integrated barometric pressure sensors and temperature sensors [The Royal Society for the Prevention of Accidents (RoSPA)]. Some systems use an accelerometer to detect the centrifugal acceleration caused by the movement of the tyre from rest there by activating the sensor [Wei, Zhou, Wang et al. (2012)]. Other systems rely on a low frequency

1Research Scholar, School of Mechanical and Building Sciences (SMBS), Vellore Institute of Technology Chennai campus, Chennai, India. Email: anoopps.vit@gmail.com
[2]Associate Professor, School of Mechanical and Building Sciences (SMBS), Vellore Institute of Technology Chennai campus, Chennai, India. Email: v_sugu@yahoo.com

receiver which receives an activation signal from the main TPMS control unit which would be housed in the vehicle. The sensor is protected within a strong plastic case and may be arranged at the neck of the tyre or may be strapped to the rim of the tyre using a steel band. A common defect reported by direct TPMS users was that the valve stem would corrode and would break with the slightest amount of torque. Moreover, direct TPMS cannot be used in conjunction with self-sealing solvents as it clogs the port of the barometric pressure sensor, rendering the sensor non-operational until cleaning. Indirect tyre pressure monitoring systems do not depend on direct pressure information; however, they rely on other indirect parameter such as wheel speeds [NIRA Dynamics AB and Dunlop Tech GmbH]. Since indirect TPMS systems are relative in nature they must be reset once the tyres are pressurized to optimum pressure to 'Re-learn' the required parameters. The 'Re-learning' procedure may take 5 to 60 minutes of driving. Indirect TPMS requires the vehicle to be in motion to operate [NIRA Dynamics. TPI].

Many studies had been carried out in direct and indirect tyre pressure monitoring system, to name a few, Howard et al. [Howard, NcGinnis and Daugherty (1993)] proposed a system in which remote the accelerometer would be fixed on the tyre and the tyre would be struck with a hammer. The accelerometer would be connected to a signal analyzer. The author presented a remote tyre pressure monitoring approach. Based on the air pressure the vibration frequency peaks generated by the tyre would alter. Hill et al. [Hill, Malson and Turner (2000)] proposed two methods for indirect tyre pressure measurement, A moving piston system and an inductive powered transducer method. However the author(s) also reported that the the vehicle had to be in motion for the system to work. Craighead [Craighead (1996)] proposed a sense tyre pressure, wheel balance and damper condition from vibration measurements. These parameters would be supervised when the vehicle is driven in a normal way. It was reported that the tyre pressure could be predicated only up to an accuracy of 12% and that further research was needed to improve the classification accuracy. Wei et al. [Wei, Zhou, Wang et al. (2012)] carried out a research on algorithms for tyre pressure monitoring based on multiple sensor information fusion and Bayesian. It was concluded that the information sensor fusion was better than the Bayesian method. Knawar et al. **[**Singh, Bedekar, Taheri et al. (2012)] proposed a piezo electric vibration harvesting system. This system uses a sensor to generate electrical power from vibration harvested from a running tyre there by powering tyre pressure monitoring sensors. Hamed et al. [Hamed, Tesfa, Aliwan et al. (2013)] carried out a research on the influence of vehicle tyre pressure on suspension system. The response of the suspension system was analyzed by applying the time – frequency approach. The vertical vibrations generated by the system were used for data analysis. Carcaterran et al. [Carcaterran and Roveri (2013)] investigate tyre grip identification based on strain information where the tyre road grip condition could be estimated from the strain information acquired from the tyre. Dubois et al. [Dubois, Cesbron, Yin et al. (2013)] conducted a study in which low frequency statistical estimations of rolling noise of tyre and road were calculated numerically. Genovesi et al. [Genovesi, Monorchio and Saponara (2008)] designed a double loop antenna for tyre pressure monitoring system. The author claimed that it could be used to reduce complexity of sensing nodes. A deflated tyre would increase its contact path to the ground [Paine, Griffiths and Magedara (2007)]. This could add rolling resistance. Extreme under-inflation can even lead to

thermal and mechanical stress caused by overheating and subsequent, sudden destruction of the tyre itself [Mohsenimanesh, Ward and Gilchrist (2009)]. Under-inflation can severely affect fuel efficiency and tyre wear. Tyres lose air overtime due to natural diffusion; even a new, properly mounted tyre can lose pressure up to 1 psi a month [Persson (2005)]. Ivan et al. [Ivan, Carlos, Holly et al. (2011)] highlighted the challenges in developing a cost effective system-in-package for tyre pressure monitoring system. A tyre maintained at optimum pressure could provide the user the benefits such as improved fuel economy, extended tyre life, decreased downtime and maintenance and improved handling. Hence, there is a strong need to propose a fault diagnosis system which can monitor the tyre pressure using machine learning approach [Anoop, Sugumaran and Praveen (2017)].

Numerous works were carried out using distinctive methodologies; however, only a very few in the experimental analysis for tyre pressure monitoring Machine learning technique was presently considered for tyre pressure monitoring; however, the usage was limited in the literature. Hence, there is a strong need to design a fault diagnosis system which can predict the tyre pressure using machine learning approach. This study makes a novel attempt to find a TPMS using machine learning approach Figure 1 shows the methodology of the work done. The rest of the paper is organized as follows. Section 2 presents the experimental setup and experimental procedure. In section 3, feature extraction is explained, followed by feature selection. The result of the classifier and classification accuracy of the models was discussed and the suggestion of the better model is proposed in section 4. Conclusions are presented in the final section (Section 5).

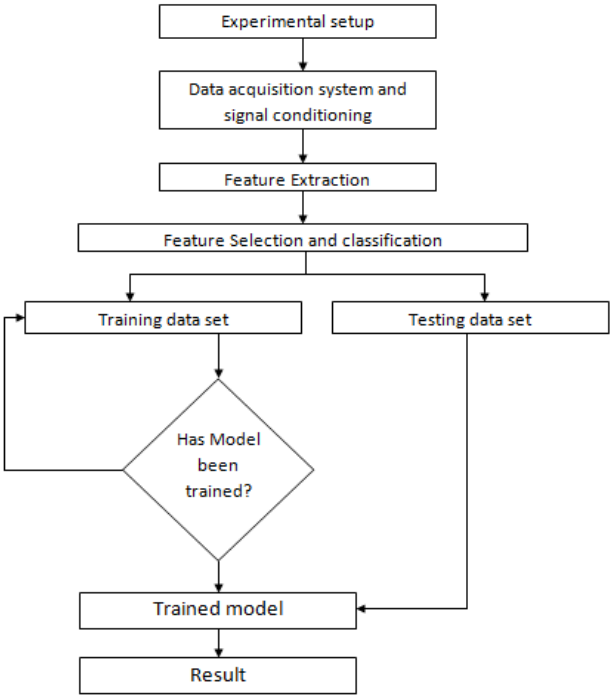

Figure 1: Methodology

2 Experimental setup

This work suggests a novel technique to monitor tyre pressure by using the machine learning and fault diagnosis approaches. A radial tyre was chosen for this study (Figure 2) because of its good fuel economy and driving comfort. Pressure readings were taken for different conditions; normal, puncture and idle. Tyre pressures of 28 psi were taken as normal and pressure readings of 22 psi and below were taken as puncture. The test speeds were limited from 20 km/h to 80 km/hr. The speed of the vehicle was varied in a normal driving method. Speeds below 20 km/h did not give sufficient amplitude hence they were classified as idle, irrespective of their tyre state. A tri-axial MEMS accelerometer was used to acquire the vibration data. Figure 3 represents the MEMS accelerometer module. Figure 4 illustrates the axes of the accelerometer. Figure 5 represents the experimental setup. Table 1 shows the MEMS accelerometer specification and Table 2 shows the specification of DAQ system used for MEMS sensor. A total of 360 samples of 1000 data points each at a sampling rate of 66Hz were taken. Equal numbers of samples were acquired for all three classes in order to ensure the experiment unbiased. The classes were titled as 'Normal', 'Puncture' and 'Idle'. A data acquisition program written in visual C++ was used to log the incoming data. The accelerometer module was coated with a waterproof gum and fixed to the axel of the rear right wheel of the car (Figure 5). A shielded wire was used to minimize external electronic interference [Anoop, Sugumaran and Praveen (2016)]. Sri et al. [Sri, Vetrivel, Mathew et al. (2015)] presented systematic analysis has been carried to optimize the sensitivity. A shielded wire was used to minimize external electronic interference. According to Nyquist Shannon sampling theorem, the minimum sampling rate should be at least double the highest incoming frequency in order to avoid antialiasing effect [McLean, Alsop and Fleming (2005)]. Hence the minimum sampling rate must be 28.26 Hz. The sampling rate was set at 66 Hz.

Figure 2: Tyre used for experiments

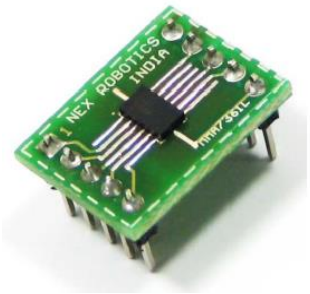

Figure 3: MEMS accelerometer module

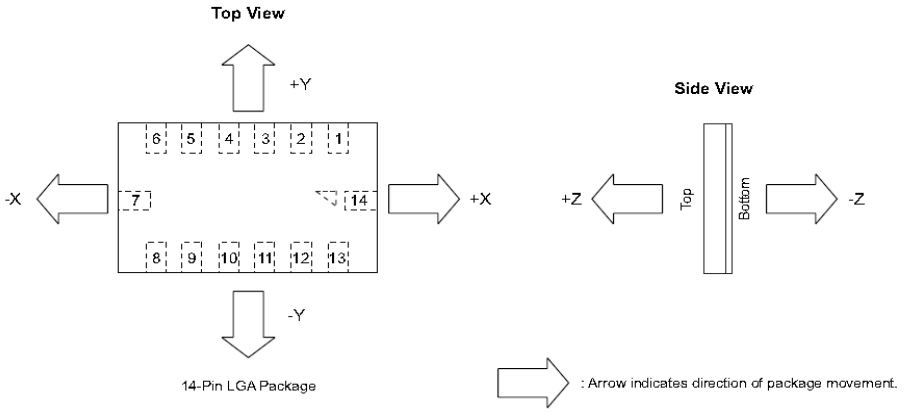

Figure 4: MEMS Accelerometer axis (from datasheet)

Figure 5: MEMS accelerometer fixed on the axel

Table 1: MEMS accelerometer specification

FEATURES	SPECIFICATION
Make	Freescale Semiconductor
Weight	<1 g (accelerometer only)
	5 g with supporting electronics
Type	MEMS
Number of Axis	3
Description	±1.5 g
	±6 g
	Selectable range
Frequency	1 – 400 Hz (X and Y axis)
	1 – 300 Hz (Z axis)
Resonance Frequency	6 kHz (X and Y axis)
	3.4 kHz (Z axis)
Sensitivity	800 mV/g @ 1.5 g
	206 mV/g @ 6 g
Connector	LGA-14 Package
	(SMD component)

Table 2: Specification of DAQ system used for MEMS sensor

FEATURES	SPECIFICATION
Make	Self
PC communication	USB/RS232
Number of input Channel	3
ADC Type	Successive approximation
ADC resolution	10 bit
Max sampling rate	15 kHz

3 Machine learning

Machine learning is a field of computer science which deals with the design and development of algorithms to develop a prediction from a supplied data set. In machine learning a signal will be represented in the form of statistical or histogram features. The selection of the right feature type for a specific application requires experimentation. Hence this paper reports a study conducted in which the same data set was processed with two different feature types. Once processed the results generated were compared.

3.1 Statistical feature processing

3.1.1 Feature extraction

Vibration signals were acquired for normal as well as puncture conditions. Different statistical features like minimum, maximum, standard deviation, skewness etc. were extracted from the acquired vibration signals. Among them the most contributing and prominent features were selected for feature selection process.

3.1.2 Feature selection using decision tree

In the feature selection, the selected features were tested by using decision tree. The contribution of each selected features were tested and the features contributing the most were considered. The remaining features were rejected to reduce computational time and load.

A standard decision tree consists of a number of branches,, nodes, leaves and one root. 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 information about the importance of the associated attribute [Peng, Flach, Brazdil et al. (2002)]. A decision tree is a tree based knowledge representation methodology used to represent classification rules. J48 algorithm (A WEKA implementation of c4.5 Algorithm) is a widely used one to construct decision trees [Quinlan (1996)]. The procedure of forming the decision tree and exploiting the same for feature selection is characterized by the following:

1. The set of features available at hand 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 being classified.
3. The branches of the tree represent each possible value of the feature node from which they originate.
4. The decision tree can be used to classify feature vectors by starting at the root of the tree and moving through it until a leaf node, which provides a classification of the instance, is identified.
5. At each decision node in the decision tree, one can select the most useful feature for classification using appropriate estimation criteria. The criterion used to identify the best feature invokes the concepts of entropy reduction and information gain-discussed in the following sub section.

Figure 6 represents the decision tree generated using the J48 tree algorithm.

3.1.2.1 Information gain and entropy reduction

Information gain measures how well a given attribute separates the training examples according to their target classification. The measure is used to select among the candidate features at each step while growing the tree. Information gain is the expected reduction in entropy caused by partitioning the examples according to the given feature.

Information gain (*S, A*) of a feature *A* relative to a collection of examples *S*, is defined as

$$Gain(S, A) = Entropy\ (S) - \sum_{v \in Value(A)} \frac{|S_v|}{|S|}\ Entropy\ (S_v), \quad (1)$$

Where *Value(A)* is the set of all possible values for attribute *A*, and S_v is the subset of *S* for which feature *A* has value v (i.e. $S_v = \{s \in S\ |A(s) = v\}$). Note the first term in the equation for Gain is just the entropy of the original collection *S* and the second term is the expected value of the entropy after *S* is partitioned using feature *A*. The expected entropy described by the second term is simply the sum of the entropies of each subset S_v , weighted by the fraction of examples $|S_v|/|S|$ that belong to S_v *Gain (S, A)* is therefore the expected reduction in entropy caused by knowing the value of feature *A*. Entropy is a measure of homogeneity of the set of examples and it is given by

$$Entropy\ (S) = \sum_{i=1}^{c} -P_I log_2 P_i \quad (2)$$

Where c is the number of classes and P_i is the proportion of *S* belonging to class *i.* [Sugumaran and Ramachandran (2007)]

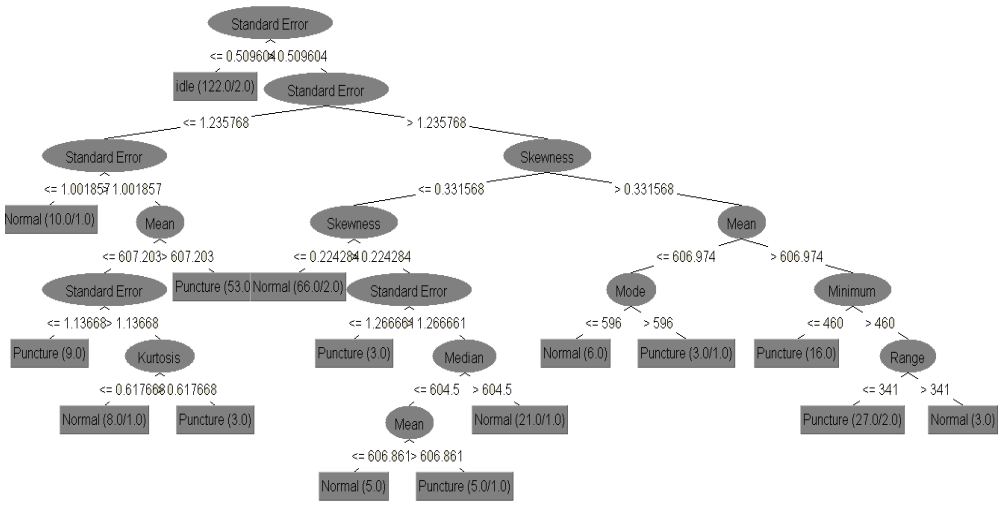

Figure 6: Decision tree generated using the J48 tree algorithm

Number of Leaves:	16
Size of the tree:	31

Time taken to build model: 0.01 seconds

Table 3: Detailed accuracy by class for the untrained J48

TP	FP	PR	R	F	RO	C
0.858	0.067	0.866	0.858	0.862	0.919	N
1	0.008	0.984	1	0.992	0.993	P
0.85	0.071	0.857	0.85	0.854	0.889	I
0.903	0.049	0.902	0.903	0.902	0.933	W

Table 4: Cross-validation for j48

Parameter	Value/ Action
Correctly Classified Instances	**325/360**
Kappa statistic	0.8542
Root mean squared error	0.2463
Root relative squared error	52.2453 %
Incorrectly Classified Instances	35/360
Mean absolute error	0.0809
Relative absolute error	18.1921 %
Total Number of Instances	360

Table 3 shows the detailed accuracy by class for the untrained J48 classifier. Table 4 shows the stratified cross-validation details for the untrained J48 classifier. Table 5 shows the confusion matrix generated by the untrained J48 classifier. Table 6 shows the values for objects of the trained J48 tree classifier.

Table 5: Confusion matrix generated the untrained J48 tree

Classified as	Normal	Puncture	Idle
Normal	102	16	2
Puncture	17	103	0
Idle	0	0	120

Table 6: Values for objects of the trained J48 tree

Sl no.	Objects	Value
1	Confidence factor	0.25
2	Minimum number of objects	2
3	Number of folds	3
4	Seed	1

From this process, the features selected were standard error, mean, skewness, kurtosis and median. The remaining attributes were rejected. ‘Minimum’ and ‘range’ were rejected even though selected by the classifier as it was found that the classification

accuracy would drop (by 1%) if used. Classification accuracy of 88.78% was attained for the untrained J48 classifier. 102/120 samples were correctly classified as normal, 103/120 samples were correctly classified as puncture, and 120/120 samples were correctly classified as idle. The diagonal elements of the confusion matrix denote the correctly classified classes. However, 17/120 samples of the class 'puncture' was incorrectly classified as normal which is about 14.16%. Moreover 16/120 samples of the class 'Normal' were incorrectly classified as puncture which is about 13.33 %. Adding to the previous misclassification two samples of normal were misclassified as idle this was about 1.81%. This indicated that a study is required in reducing misclassification thereby increasing the classification accuracy [Anoop, Sugumaran and Praveen (2016)].

3.2 Histogram feature processing

3.2.1 Feature selection

In feature selection process the features extracted from the feature extraction process will be tested to determine its contribution and the features with the best contribution would be selected, others would be rejected to reduce computational load. Here J48 classifier was used to accomplish the task of feature selection as it gave the best accuracy. The decision tree generated by the classifier is shown in previous papers [Anoop, Sugumaran and Praveen (2016)]. All 99 bins were tested with the J48 classifier. Out of which, bin number 32 yielded the maximum result of 88.18%. Hence bin number 32 was chosen for the feature selection process. From this process the features selected were H9, H15, H16, H18, H19, H20, H21, H22, H24 and H25. These attributes were chosen from all the other features by the classifier and the remaining attributes were rejected.

Table 7: Detailed accuracy by class for the untrained J48 classifier for J48 tree

TP	FP	PR	R	F	RO	C
0.8	0.059	0.871	0.8	0.834	0.875	N
0.864	0.091	0.826	0.864	0.844	0.886	P
1	0.018	0.965	1	0.982	0.988	I
0.888	0.056	0.887	0.888	0.887	0.917	W

Table 7 shows the detailed accuracy by class for the untrained J48 classifier. Table 8 shows the stratified cross-validation details for the untrained J48 classifier. Table 9 shows the confusion matrix generated by the untrained J48 classifier. Table 10 shows the values for objects of the trained J48 tree classifier.

Table 8: Cross-validation for J48

Parameters	**Results**
Correctly Classified Instances	**293/330**
Kappa statistic	0.8318
Root mean squared error	0.2647
Root relative squared error	56.1433%
Incorrectly Classified Instances	37/330
Mean absolute error	0.0888
Relative absolute error	19.9839%
Total Number of Instances	330

Table 9: Confusion matrix generated J48 tree

Classified as	**Normal**	**Puncture**	**Idle**
Normal	88	20	2
Puncture	13	95	2
Idle	0	0	110

Table 10: Values for objects of the trained J48 tree

Sl no.	Objects	Value
1	Confidence factor	0.25
2	Minimum number of objects	2
3	Number of folds	3
4	Seed	1

From the above results it is clear that bin 32 had attained the maximum classification accuracy of 88.78%. 88/110 samples were correctly classified as normal, 95/110 samples were correctly classified as puncture, and 110/110 samples were correctly classified as idle. The diagonal element of the confusion matrix contains correctly classified instances. However, 13/110 samples of the class ‘puncture’ was incorrectly correctly classified as normal which is about 11.80%. Moreover 20/110 samples of the class ‘Normal’ were incorrectly correctly classified as puncture which is about 18.18%. Adding to the previous misclassification 2 samples of each of normal and puncture were misclassified as idle which is about 1.81% per class. This clearly indicates that a study is required to reduce the misclassification percentage thereby increasing the classification accuracy.

4 Results and discussion

4.1 Classification using statistical features and logistic model tree

The classifier is used for building 'logistic model trees' (LMT), which are classification trees with logistic regression functions at the leaves. The algorithm can deal with binary

and multi-class target variables, numeric and nominal attributes and missing values. Table 11 shows the confusion matrix of the trained classifier. Table 12 shows the stratified cross-validation details. Table 13 shows the detailed accuracy by class.

Table 11: Confusion matrix for LMT

Classified as	Normal	Puncture	Idle
Normal	106	13	1
Puncture	13	107	0
Idle	0	0	120

Table 12: Cross-validation for LMT

Parameters	Results
Correctly Classified Instances	333/360
Kappa statistic	0.8875
Root mean squared error	0.208
Root relative squared error	46.97%
Incorrectly Classified Instances	27/360
Mean absolute error	0.0957
Relative absolute error	21.54%
Total Number of Instances	360

Table 13: Detailed accuracy by class for LMT

TP	FP	PR	R	F	RO	C
0.883	0.054	0.891	0.883	0.887	0.948	N
0.892	0.054	0.892	0.892	0.892	0.943	P
1	0.004	0.992	1	0.996	0.998	I
0.925	0.038	0.925	0.925	0.925	0.963	W

The classifier depends on three variables which are minimum number of instances, number of boosting iterations and weight trim. The variation of these parameters *vs.* the algorithms classification accuracy is plotted in Figure 7-9 respectively.

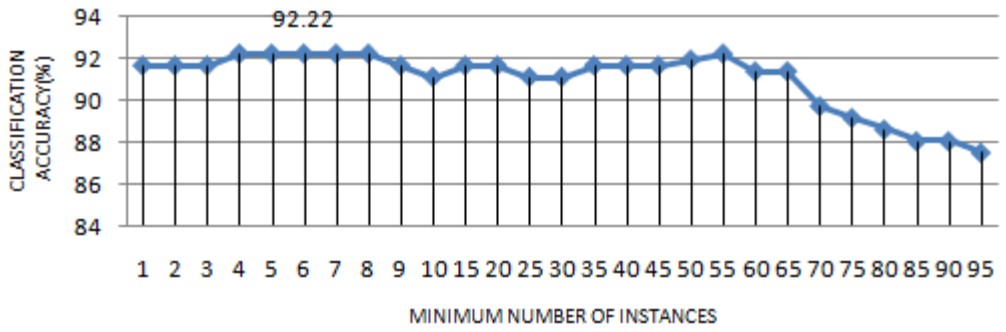

Figure 7: Minimum number of instances *vs.* classification accuracy

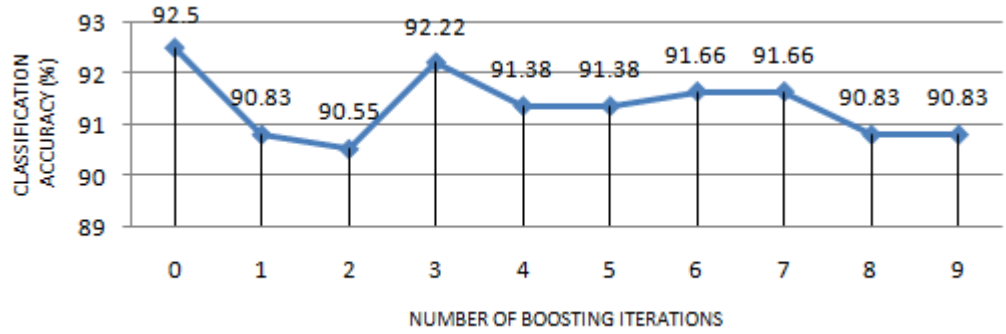

Figure 8: Number of boosting iterations *vs.* classification accuracy

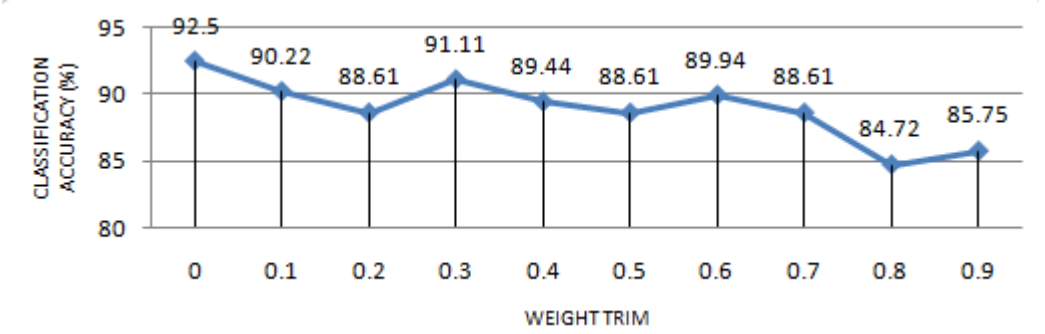

Figure 9: Weight trim *vs.* classification accuracy

Table 14: Values for objects of the trained LMT

Sl no.	Objects	Value
1	Minimum number of instances	2
2	Number of boosting iterations	0
3	Weight trim	0.1

Varying the parameter titled 'minimum number of instances' (Figure 7) from 1 to 100 in steps of '5' does not cause any change in classification accuracy initially, however on further alteration the classification accuracy increases and then drops. Varying the parameter titled 'Number of boosting iterations' from 0 to 9 in steps of '1' caused a fluctuation in classification accuracy. However, it did not improve the classification accuracy (Figure 8) which was maintained, varying the parameters 'weight' (Figure 9) from 0 to 0.9 in steps of '0.1'caused a steady drop in classification accuracy. The values of objects for the trained algorithm are shown in Table 14. From the confusion matrix (Table 11) it can be noted that 106/120 samples were correctly classified as normal, 107/120 samples were correctly classified as puncture, and 120/120 samples were correctly classified as idle. The classifier achieved a maximum classification accuracy of 92.5% after training.

4.2 Classification using histogram and logistic model tree

Table 15 shows the confusion matrix of the trained classifier. Table 16 shows the stratified cross-validation details. Table 17 shows the detailed accuracy by class.

Table 15: Confusion matrix for LMT

Classified as	**Normal**	**Puncture**	**Idle**
Normal	92	13	5
Puncture	13	95	2
Idle	0	0	110

Table 16: Cross-validation for LMT

Parameters	**Results**
Correctly Classified Instances	297/330
Kappa statistic	0.85
Root mean squared error	0.2478
Root relative squared error	52.56%
Incorrectly Classified Instances	33/330
Mean absolute error	0.0942
Relative absolute error	21.19%
Total Number of Instances	330

Table 17: Detailed accuracy by class for LMT

TP	**FP**	**PR**	**R**	**F**	**RO**	**C**
0.836	0.059	0.876	0.836	0.856	0.913	N
0.864	0.059	0.88	0.864	0.872	0.931	P
1	0.032	0.94	1	0.969	0.992	I
0.9	0.05	0.899	0.9	0.899	0.945	W

The classifier depends on three variables which are minimum number of instances, number of boosting iterations and weight trim. The variation of these parameters *vs.* the algorithms classification accuracy is plotted in Figure 10-12 respectively.

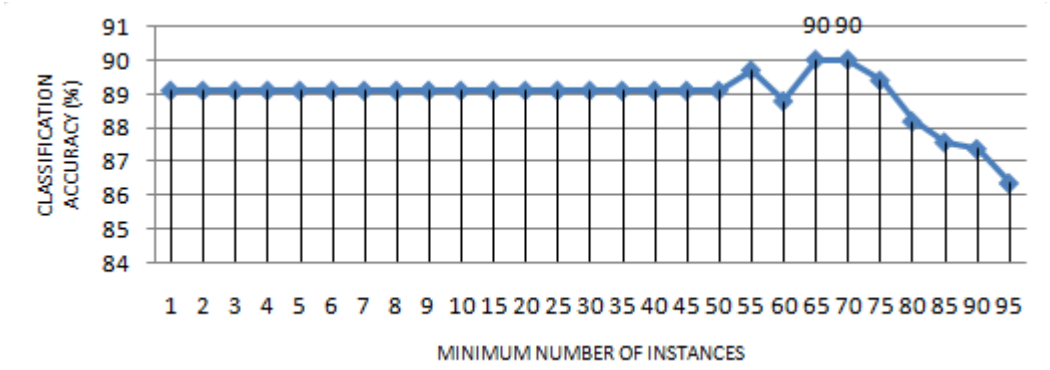

Figure 10: Minimum number of instances *vs.* classification accuracy

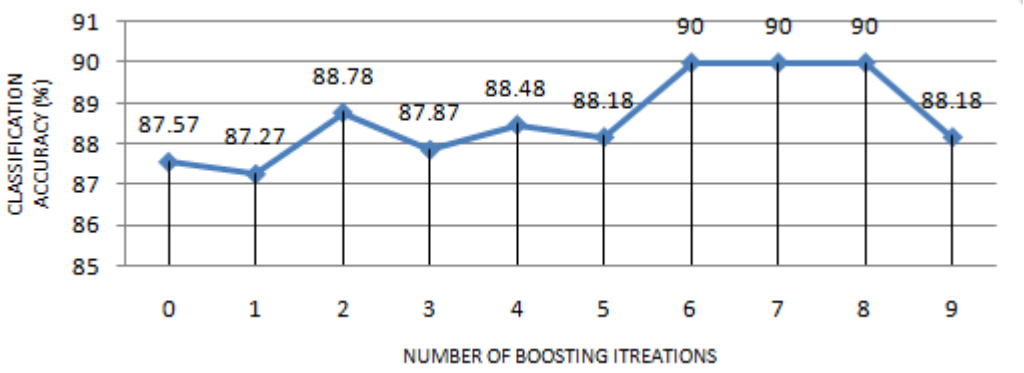

Figure 11: Number of boosting iterations *vs.* classification accuracy

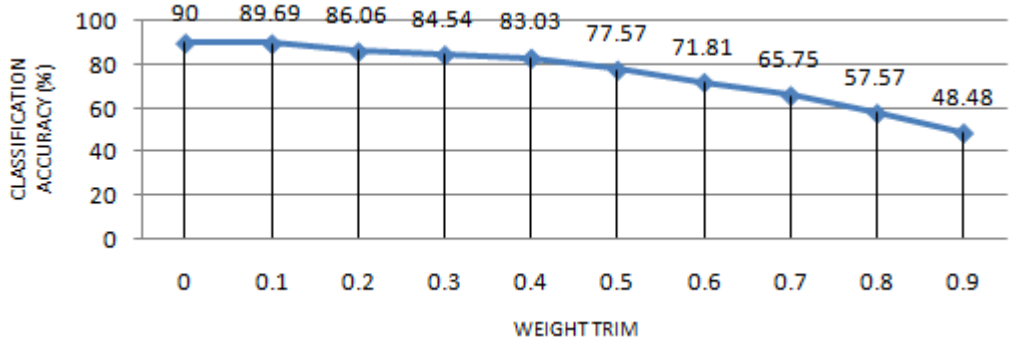

Figure 12: Weight trim *vs.* classification accuracy

The values of objects for trained algorithm are shown in Table 18

Table 18: Values for objects of the trained LMT

Sl no.	Objects	Value
1	Minimum number of objects	65
2	Number of boosting iterations	6
3	Weight trim	0.0

In existing work the classifier used was K star algorithm. Varying the parameter 'global blend from 5 to 100 in steps of '5' causes a steady drop in classification accuracy; the default value provided the best classification accuracy. As mentioned above all four modes were tested and yielded the same result. From the confusion matrix it can be noted that 92/110 samples were correctly classified as normal, 96/110 samples were correctly classified as puncture, and 110/110 samples were correctly classified as idle. The classifier achieved a maximum classification accuracy of 87.87% after training [Anoop, Sugumaran and Praveen (2016)].

Using Logistic Model Tree algorithm varying the parameter titled 'minimum number of instances' (Figure 10) from 1 to 100 in steps of '5' does not causes any change in classification accuracy initially, however on further alteration the classification accuracy increases and then drops. Varying the parameter titled 'Number of boosting iterations' from 1 to 9 in steps of '1' caused a fluctuation in classification accuracy; however, did not improve the classification accuracy (Figure 11). Varying the parameters 'weight' (Figure 6) from 0 to 0.9 in steps of '0.1' caused a steady drop in classification accuracy. The values of objects for the trained algorithm are shown in Table 18. From the confusion matrix (Table 15), it can be noted that 92/110 samples were correctly classified as normal, 95/110 samples were correctly classified as puncture, and 110/110 samples were correctly classified as idle. The classifier achieved a maximum classification accuracy of 90% after training. So the proposed Logistic Model Tree algorithm having more accurate than the existing K star algorithm.

5 Conclusion

The tyre pressure monitoring system is very important in vehicle safety. "This paper proposes a machine learning approach where the vertical vibration of a wheel hub was used to monitor the tyre pressure. The model was tested in 10-fold cross-validation. The logistic model tree was used and the highest classification accuracy of 92.5% was obtained by 10-fold cross-validation. The classification accuracy is high compared to the existing work. Hence, the Bayes Net can be practically used for the condition monitoring of wind turbine blade to reduce the downtime and to maximize the usage of wind energy. The methodology and algorithm suggested in this paper can be potentially used for any kind of tyre pressure monitoring system with minimal modification. This can ensure that a real-time tyre pressure monitoring system would be made possible if the logistic model tree is implemented.

Reference

Anoop, P. S.; Sugumaran, V.; Praveen, H. M. (2016): Implementing K-Star Algorithm to Monitor Tyre Pressure Using Extracted Statistical Features from Vertical Wheel Hub Vibrations. *Indian Journal of Science and Technology,* vol. 9, no. 47, pp. 1-7**.**

Anoop, P. S.; Sugumaran, V.; Praveen, H. M. (2017): Tyre Pressure Monitoring System Using Machine Learning Approaches-A Review. *International Journal of Control Theory and Applications,* vol. 51, pp. 371-382.

Carcaterran, A,; Roveri, N. (2013): Tire grip identification based on strain information: Theory and simulations. *Mechanical Systems and Signal Processing*, vol. 41, pp. 564-580.

Craighead, I. A. (1996): Sensing tyre pressure, damper condition and wheel balance from vibration measurements. *Proc institution of Mechanical Engineers,* vol. 211, pp. 257-265.

Dubois, G.; Cesbron, J.; Yin, H. P.; Anfosso-Lédée, F.; Duhamel, D. (2013): Statistical estimation of low frequency tyre/road noise from numerical contact forces. *Applied Acoustics*, vol. 74, pp. 1085-1093.

Genovesi, S.; Monorchio, A.; Saponara. S. (2008): Double-loop antenna for wireless tyre pressure monitoring. *Electronic Letters,* vol. 44, no. 24, pp. 1385-1386.

Hamed, M.; Tesfa, B.; Aliwan, M.; Li, G.; Gu, F. et al. (2013): The Influence of Vehicle Tyres Pressure on the Suspension System Response by Applying the Time-Frequency Approach. *International Conference on Automation & Computing,* pp. 1-6.

Hill, M.; Malson, P. R. W. C.; Turner, J. D. (1990): The development of a low cost system for monitoring tyre pressures. *Chassis Electronics*, pp. 1-3.

Howard, H. R.; NcGinnis, T. A.; Daugherty, R. H. (1993): Remote tire pressure sensing technique. *NASA Case*, vol. 141, pp. 60-61.

Ivan, M. L. S.; Carlos, S. W. C.; Holly, T. C. L.; Lam, P. M.; Yeung, J. Y. et al. (2011): Challenges in developing cost-effective system-in-package. (sip) for tyre pressure monitoring system (TPMS). *International Conference on Electronic Packaging Technology & High Density Packaging*, pp. 1-6.

McLean, R. F.; Alsop, S. H.; Fleming, J. S. (2005): Nyquist-overcoming the limitations. *Journal of Sound and Vibration,* vol. 280, pp. 1-20.

Mohsenimanesh, A.; Ward, S. M.; Gilchrist, M. D. (2009): Stress analysis of a multi-laminated tractor tyre using non-linear 3D finite element analysis. *Materials and Design*, vol. 30, pp. 1124-1132.

NIRA Dynamics, A. B.; Dunlop Tech GmbH. (2015): *Indirect Tire Pressure Monitoring Systems-Myths and Facts.*

http://www.businesswire.com/news/home/20120720005231/en/Nira-Dynamics-Indirect-Tire-Pressure-Monitoring-Systems%20#VRUBp47ANhk.

NIRA Dynamics. (2015): TPI-Advanced Indirect Tire Pressure Monitoring. NIRA Dynamics AB. www.niradynamics.se/scripts/resource.

Paine, M.; Griffiths, M.; Magedara, N. (2007): The role of tyre pressure in vehicle safety, injury and environment. *Road safety solutions.*

http://www.maic.qld.gov.au/forms-publicationsstats/pdfs/tyre-pressure-report-final.pdf.

Peng, Y. H.; Flach, P. A.; Brazdil, P.; Soares, C. (2002): Decision tree-based data characterization for meta-learning. *ECML/PKDD-2002 workshop IDDM-2002.*

Persson, N. (2005): Estimation properties of a tire pressure monitoring system. *Link opings university*, vol. SE-581 83 Link oping, pp. 1-6.

Quinlan, J. R. (1996): Improved use of continuous attributes in C4.5. *Journal of Artificial Research,* vol. 4, pp. 77-90.

Singh, K. B.; Bedekar, V.; Taheri, S.; Priya, S. (2012): Piezoelectric vibration energy harvesting system with an adaptive frequency tuning mechanism for intelligent tires. *Mechatronics*, vol. 22, pp. 970-988.

Sri, R. N.; Vetrivel, S.; Mathew, R.; Sankar, A. R. (2015): Effect of electroplated gold film on the performance of a piezoresistive accelerometer with stress concentrated tiny beams. *Indian Journal of Science and Technology*, vol. 8, pp. 1-5.

Sugumaran, V.; Ramachandran, K. I. (2007): Automatic rule learning using decision tree for fuzzy classifier in fault diagnosis of roller bearing. *Mechanical Systems and Signal Processing*, vol. 21, pp. 2237-2247.

The Royal Society for the Prevention of Accidents (RoSPA). (2015): *Road safety information*.

http://www.rospa.com/road-safety/advice/vehicles/tyre-safety-technology/pressure-monitoring-systems.

Velupillai, S.; Güvenç, L. (2007): Tire Pressure Monitoring. *IEEE Control Systems Magazine*, vol. 1, pp. 275-280.

Wei, C.; Zhou, W.; Wang, Q.; Xia, X.; Li, X. (2012): TPMS (tire-pressure monitoring system) sensors: Monolithic integration of surface-micro machined piezo resistive pressure sensor and self-testable accelerometer. *Microelectronic Engineering*, vol. 91, pp. 167-173.

Yulan, Z.; Yanhong, Z.; Yahong, L. (2012): Based on Multi-sensor Information Fusion Algorithm of TPMS Research. *International Conference on Solid State Devices and Materials Science, Physics Procedia*, vol. 25, no. 5, pp. 786-792.