International Journal of Natural Sciences Research
2016 Vol. 4, No. 3, pp. 56-67
ISSN(e): 2311-4746
ISSN(p): 2311-7435
DOI: 10.18488/journal.63/2016.4.3/63.3.56.67

DETECTION OF MOTORWAY DISORDERS BY PROCESSING AND CLASSIFICATION OF SMARTPHONE SIGNALS USING ARTIFICIAL NEURAL NETWORKS

Yusra Mohammed M. Salih[1]† --- **Ali Kattan**[2] --- **Taner Çevik**[3]
[1,3]*Computer Engineering, Fatih University, Istanbul, Turkey*
[2]*IT Department, ISHIK University, Erbil, Iraq*

ABSTRACT

Potholes, debris, sunken manhole covers and others are common street safety hazards which drivers experience daily as they bump into them unexpectedly while driving. Pavement roughness is usually evaluated based on the International Roughness Index (IRI), which is considered the most prevalent metric. In this work, IRI values are collected by using a smartphone, with built-in vibration sensor, placed on the car's dashboard while driving around the city. The classification process of IRI values is primarily performed using an Artificial Neural Network (ANN) for the detection of diverse predefined street safety hazards. The designed ANN is a backpropagation pattern classifier, that must be trained to yield either a detected "disorder" area of the road or "normal" area based on the IRI data collected. The process of preparing the training and testing datasets involves a number of pre-processing operations. The IRI values are pre-processed in order to extract the most effective features. Then the network is trained with the normalized feature set by using supervised learning method. The performance of the designed network is compared to a similar works in the literature. Results show that the designed network can successfully classify the street conditions by using IRI values with a success rate that outperforms the classification rates obtained by other works.

Keywords: Road safety, Smartphone, International roughness index, Artificial neural network, Backpropagation, Supervised training.

Received: 15 August 2016/ **Revised:** 31 August 2016/ **Accepted:** 16 September 2016/ **Published:** 21 October 2016

1. INTRODUCTION

Road roughness, also referred to as smoothness, is an important pavement characteristic. It affects not only driving quality but also vehicle performance, fuel consumption and maintenance costs. Road maintenance is a vital and challenging service that city municipals must handle carefully. Potholes, debris and sunken manhole covers are some of the common road disorders that poses safety risks and also sources of complaints by drivers [1]. Detecting the road condition has gained significant importance in the last few years. There are various reasons for attracting the focus of the researchers to this field. Firstly, it will ensure safety and comfort and secondly it will save considerable costs related to maintenance as problems can be addressed as soon as possible. Smooth and safe roads will lead to less vehicle damage and government investment [2].

Various proposed techniques require dedicated hardware installed in vehicles or at several road joints [3-5]. These methods typically focus on roads of developed countries that owe relatively simple road and traffic flow patterns. Furthermore, these methods are not cheap since they require specialized hardware, time and effort. Attaching sensors to a large number of vehicles and various road joints is impractical due to the great financial cost and human effort involved.

In the past few years, the widespread of low-cost smartphones and globally prevalent Internet access through these devices have inspired developers to write promising applications [6]. Large memory capacity, powerful

† Corresponding author

processing capabilities, high-speed internet connectivity through 3G/4G or Wi-Fi, various built-in sensors such as an accelerometer, magnetometer, gyroscope, microphone, camera, GPS to sense and capture data from the environment are the prominent specifications that make smartphones an integral part of our daily lives [7, 8]. The simplicity of using smartphones and the availability of on-the-move Internet access have led application developers to develop applications which can access real-time information offered by Cloud Servers.

The smartphone's built-in sensors utilization is proposed in several works to detect road hazards and conditions [9-11]. The established method using smartphones eliminates the need for installing special sensors in vehicles. Smartphones became cheap commodities as newer, and cheaper models with greater capabilities are being released [12]. These features make smartphones better candidates to use for detecting and analyzing road hazards. The concept also allow for scalability as the number of mobile users increases and users themselves play a potential role in collecting and sharing data about the road conditions.

However, some sort of software intelligence is needed to identify whether the detected road condition is really a road hazard or normal driving maneuver performed by the driver. The smartphone application should comprise machine learning techniques to be able to detect such disorders. By utilizing Cloud processing, that is associated with geospatial information, the application could access such a service to identify the road conditions to give the best performance results [1].

This study considers the problem of identifying and analyzing the road conditions as an essential step that would help city municipals to gather proper data, based on international standards, and take the suitable action to maintain roads. The main aim of this work is collect suitable data in order to design and train an ANN that will distinguish such road hazards. In the data gathering stage an Android smartphone application is used to collect the IRI patterns while driving on actual roads. The application uses the smartphone's accelerometer to measure and produce the IRI patterns. Then the data is analyzed, filtered and normalized in order to efficiently train an ANN using multi-layer perceptron with back-propagation algorithm to improve the identification and proper classification of road surface conditions. The resultant outcome can be used as part of a cloud-based service to facilitate the road maintenance process by associating such process with geospatial services (such as GPS) offered by the phone to indicate the location of such road hazard.

The rest of the paper is organized as follows: Section 2 is the related work whereby it covers some of the research studies that have been previously performed for detecting and analyzing road bump conditions using special instruments or vehicles. The proposed method is presented in section 3. The implementation and results are given in section 4. Discussion is given in section 5. Lastly, the conclusion and future work are in section 6.

2. RELATED WORK

Driving became an essential routine in our daily lives and has a significant effect safety and financial costs. Inherently, roads are the primary element of the driving system. Continuous rolling under the wheels, natural factors such as snow, rain and others lead to developing various impairments and anomalies on the roads which significantly affect the quality of driving.

Some measurement such as roughness is required to determine the road condition. Roughness is measured usually using some form of index values such as International Roughness Index (IRI), Present Serviceability Rating (PSR), Power Spectral Density (PSD), Riding Quality Index (RQI), etc. [13]. Several techniques are used to gather data and calculate the indices mentioned above.

An excessive amount of research activities has been proposed in this field using special instruments or vehicles [13, 14]. This includes proprietary embedded sensors in vehicles [3, 5] or by using smartphone applications that make use of the smartphone's deployed sensors for detecting road conditions [15, 16]. Machine learning plays a crucial part in analyzing such roughness index data using either special propose algorithms [4] or other machine learning techniques to identify road safety hazards [1, 17].

3. THE PROPOSED METHOD

A similar system was originally proposed in Kattan and Aboalmaaly [1]. The architecture comprises of four main steps. Data Gathering, Data Transmission via Cloud Service, Dataset Preparation, ANN Design. The proposed system is different from the one proposed in Kattan and Aboalmaaly [1] in that it uses IRI values as a standard rather than processing resultant signals directly (waves analogous to the sensor output values) and as shown in Fig. 1.

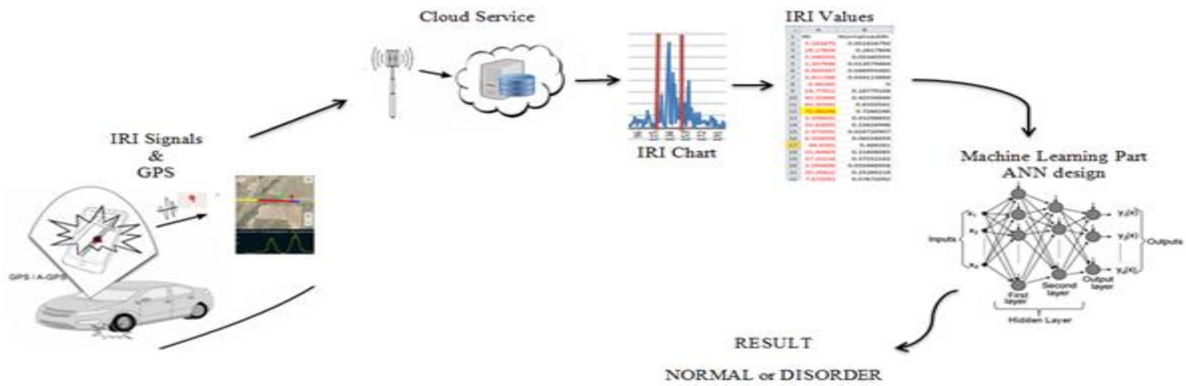


Fig-1. General overview of the proposed architecture [1]

3.1. Data Gathering

The smartphone uses an application that records IRI values while driving. This process involves driving on roads with and without anomalies. Such data is to be filtered to and the anomalies are to be isolated as "segments". Patterns are created from such data in order to prepare for the dataset. The accepted segments of IRI samples are transferred to a spreadsheet file for further processing. The IRI points for normal and disorder areas are selected to be 21 IRI values in total and become the attributes of road samples. The attribute sample for the disordered area is created by selecting the highest IRI value with ten points after and before it which will be 21 in total. For creating attribute sample for the normal area, the IRI value greater than zero and less than 6 is selected with ten points after and before it which again will be 21 points in total. Measurements for disorder and normal portions of the road are depicted in Fig.2 (a-b) respectively.

Fig-2. IRI signals for normal and disorder area.
(The line chart of the IRI (m/Km) value generated using Excel)

3.2. Dataset Preparation

The collected IRI values for normal and disorder area would have different ranges. All data should be normalized (as a real number) to be in the range [0, 1] to prepare for a suitable training and testing data for the proposed ANN design. The normalization is done by dividing each IRI value by 100 to scale the values to the range [0-1]. Since there is no vertical movement of the vehicle, the accelerometer reads the IRI value to be less than zero.

To avoid this situation, the negative IRI values in each selected normal and disorder samples are neglected and assigned to the value of zero.

After the pre-processing stage of the IRI values, the next step involves the process of creating the datasets. The collected IRI values are divided into two groups “Disorder Area” and “Normal Area”. The “Disorder Area” represents any street which has anomalies and it is assigned the value of 0. The “Normal Area” represent smooth areas of roads and it is assigned the value of 1.

3.3. ANN Design

A Feed-Forward Artificial Neural Network (FFANN) with back-propagation gradient-decent based search method has been designed for the classification of IRI pattern signals. The designed FFANN consists of three layers: input layer, log- sigmoid hidden layer and log-sigmoid output layer.

Each input is weighted with an appropriate weight. The sum of the weighted inputs and the bias forms the input to the sigmoid transfer function which is calculated using Eq. (1) [18].

$$y_i\,(x) = \frac{1}{1 + e^{-z_i}} \qquad (1)$$

Back-propagation learning algorithm is applied to the neural network for supervised training so that it can minimize the differences between the simulated and target output(s). The Gradient Descent (GD) algorithm is used for the backpropagation training momentum term, and variable learning rate is applied to speed up the BP. The training of the neural network is made as generalized to avoid overfitting [19].

According to Eq. (2), the termination of the network occurs when the total mean squared error (MSE) becomes less than or equal to 0.001 [19] or when it reaches to a different number of epochs which is specified according to the network design. For each iteration and session, the weight and bias values associated with different layers are saved automatically. If the simulation results from ANN are not satisfactory, the network has been tuned and trained again with the last saved weight and bias values. This has been done to improve the network performance and reduce time consumption for training.

$$F = mse = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(t_i - a_i)^2 \qquad (2)$$

The classification of IRI values is considered through different designs of the ANN network. For the purpose of finding an optimum number of hidden layers and corresponding neuron size for best classification performance, a different number of hidden layers with the same number of neurons has been selected for both type of the design and their classification accuracy are also reported accordingly.

The confusion matrix [20] is used to estimate the performance of the proposed ANN design. Accuracy, sensitivity, and specificity are evaluated to evaluate the performance.

True Positive Rate (TPR): Roads correctly identified as Disorder

True Negative rate (TNR): Roads correctly identified as Normal

False Negative rate (FNR): Road incorrectly identified as Normal

False Positive rate (FPR): smooth roads are mistakenly classified as a disorder.

The confusion matrix on the road analysis is given in Table 1:

Table-1. Confusion matrix for road analysis (indicating the values for normal and disorder)

Actual value	**Expected value**	
	Disorder	Normal
Disorder	TP	FN
Normal	FP	TN

The dataset is to be divided into training and testing dataset and applied to the proposed design to test the accuracy. Fig. 3 shows the training process involved for the selected set of training patterns.

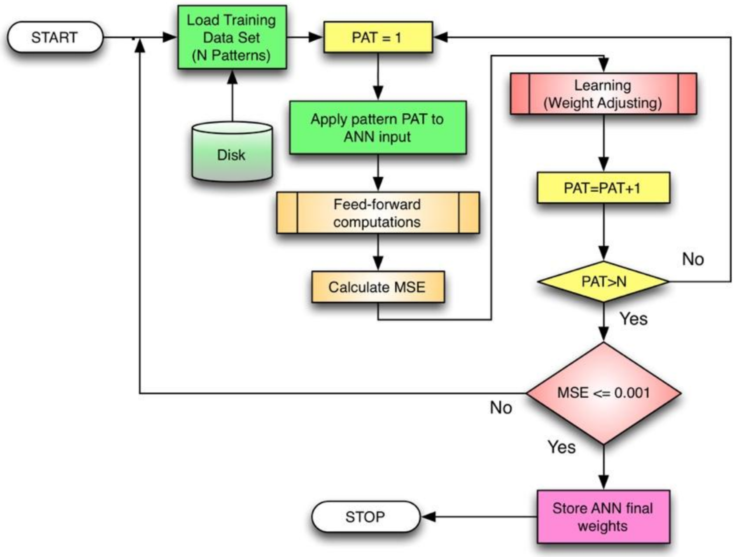


Fig-3. The ANN training process.

The proposed ANN considered two designs, single hidden layer design and two hidden layers design as shown in Fig. 4 and Fig. 5 respectively. The single hidden layer design ANN comprises 21 neurons in the input layer, each representing one of the sampled IRI points, 10 neurons in the hidden layer and two neurons in the output layer which represent the classified Disorder or Normal areas.

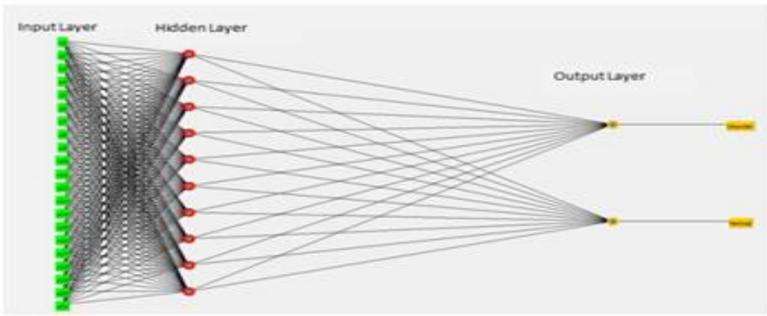


Fig-4. The structure of the designed single hidden layer ANN (generated by Weka using the proposed model)

The two hidden layer design is the same as the single layer in terms of input and output but instead has two hidden layer each with 10 hidden neurons.

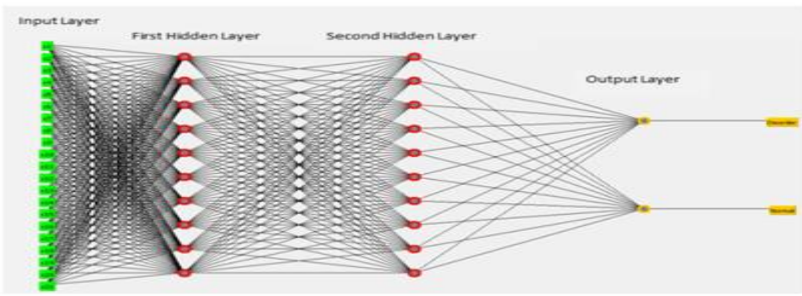


Fig-5. The structure of the designed ANN with two hidden layers (Generated by using Weka for the proposed model)

4. IMPLEMENTATION & DISCUSSION

The data is gathered by first designating some streets with anomalies in the cities of Sulaymaniyah and Erbil, at the north of Iraq. A total of 58 different types of streets were tested and some samples are shown in Fig.6. The driving speed is considered to be the speed limits set for that street, which ranged between 40 to 80 km/h. Data collection considered highway streets, main city streets, and neighborhood streets.

In order to create a proper dataset, various types of roads are considered for data collection such as crosswalks using extra thick point brick, or strips of powers and metal expansion joints in bridges and overpasses. Different types of speed bumps were also considered such as raised dots and extended dots or lines which are placed on some streets of the city by municipals to forcibly limit the driving speed in those area. Such speed bumps are to be considered as "normal" and they do not represent "anomalies". Anomalies would consider driving over things such as potholes, broken manhole covers, missing chunks of pavements severely sunk in protruding manhole covers, and other significant road surface anomalies such as moderate cracking, damaged pavements and rough unpaved roads, which occur after putting the roads into use after construction.

Fig-6. Diverse street samples (showing normal speed bumps, manholes, potholes, etc.)
Photos were taken by the others in the cities of Sulaimaniyah and Erbil (North of Iraq).

The RoadBump Pro © Android application was used to collect data and records the roughness (IRI values) of the road using the accelerometer sensor in the smartphone [21]. The application also utilizes GPS to capture the location area of the road. The produced IRI values generated by the Smartphone match the real IRI results which previously produced by very high-cost inertial profilers.

RoadBump Pro © was installed on a Samsung galaxy A5 Duos © smartphone and the smartphone is placed on the car's dashboard using a non-slipping mat and data are recorded by driving over actual road comprising normal and disorder areas with different speed considerations. A car smartphone charger was continuously plugged in to ensure not running out of battery [1].

The recorded data were equally divided into two set; Normal and Disorder data. A dataset of collective IRI patterns is created by mixing these. Each pattern would have 21 attributes of IRI sampling points with a total of 195 IRI sample patterns were prepared as described earlier in the previous section. Table 2 is a summary of the obtained dataset.

Table-2. Selected data to be used for the MLP NN design

Segment types	Number of segment samples
Disorder sample	98
Normal sample	97
Total	195

The dataset is divided into two separate datasets using percentage split. Each dataset comprises of equal percentages of "Disorder" and "Normal" samples. The Diagram in Fig.7 explains the working procedure on the datasets.

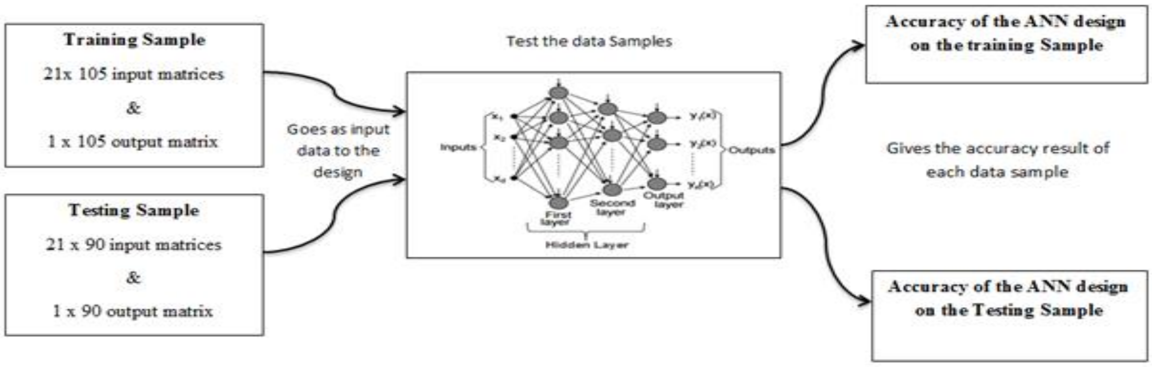
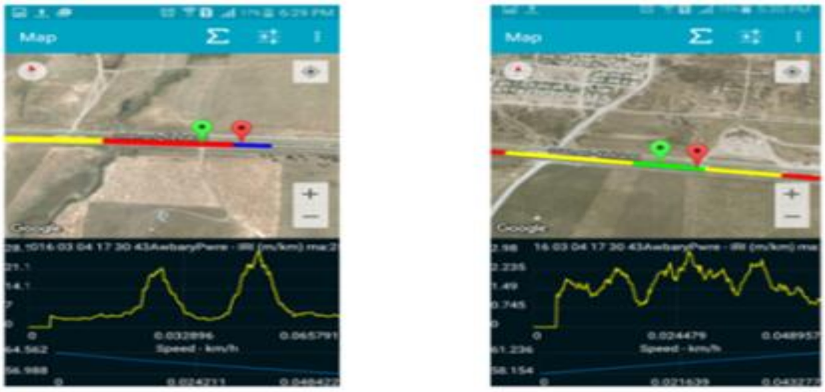


Fig-7. Working procedure on training and testing sample datasets. It shows the diagram for training & testing sample datasets on the designed neural network and testing the accuracy for both training sample and testing sample.

A map of the sampled roads with the IRI moving average graph (m/Km) is shown in Fig.8. IRI moving average graph uses a variable-sized moving average. The number of accelerometer readings being averaged into each point on the graph varies with the number of accelerometer records between the markers [22].

Fig-8. Map with IRI graph. (collected by authors) Screenshot of the application used for collecting data using IRI moving average graph which uses a variable sized average.

Long paths in each drive are subsequently filtered to save smaller sections between the *Start* and *End* markers as individual recordings. These recordings include both "Normal" and "Disorder" areas of the roads targeted as shown in Fig.9. The segments which do not satisfy the data collection considerations are neglected and others are saved to be tested in the ANN network design.



a. Disorder Segment b. Normal Segment

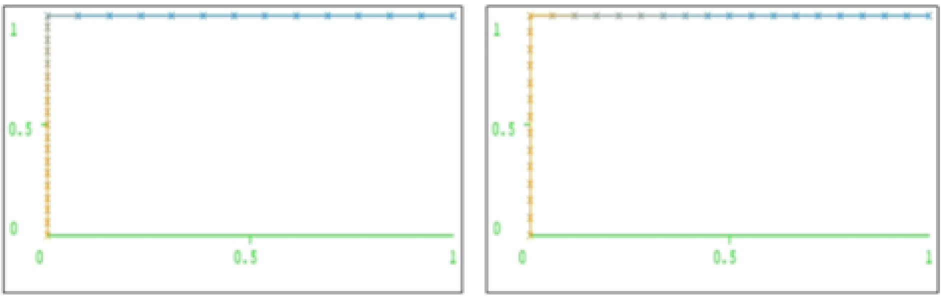
Fig-9. Targeted segments on the smartphone application. (Collected by authors) Screenshot of the mobile application used for collecting the data)

Weka © tool bench is used to create the ANNs proposed earlier and to train and use the ANN for the classification of collected dataset. Weka © is a powerful open source software, yet easy collection of machine learning algorithms for data mining tasks supplied under the GNU General Public License. The algorithms, such as SVM and multilayer perceptron, can either be applied directly to a dataset or called from our Java code for classification on big datasets [23]. Training would consider using 70% of the dataset as training set and the rest 30% as testing set.

4.1. Performance Evaluation using Single-Hidden Layer

The results show that 30 instances of the test split dataset are correctly classified with percentage 96.774 %, and one instance of the test split dataset is incorrectly classified with a mean squared error (MSE) value of 0.2094.

The ROC area under the curve of percentage test split for normal and disorder area is shown in Fig. 10.



A. ROC area for Disorder B. ROC area for Normal

Fig-10. ROC Area under the curve of percentage split test for single hidden layer ANN design. The threshold value for 30 instances of the split test set for training dataset visualized using Weka, x-axis: False positive rate and y-axis: True positive rate

To test the accuracy of the ANN design, a validation test sample is run based on the saved design and the performance evaluation is calculated. The results clarify that 87 instances out of 90 instances of user supplied test dataset are correctly classified with an accuracy of %96.667, and three instances of the supplied test dataset are incorrectly classified with a mean squared error (MSE) value of 0.1969. The ROC area under the threshold curve of disorder and normal class is shown in Fig. 11.

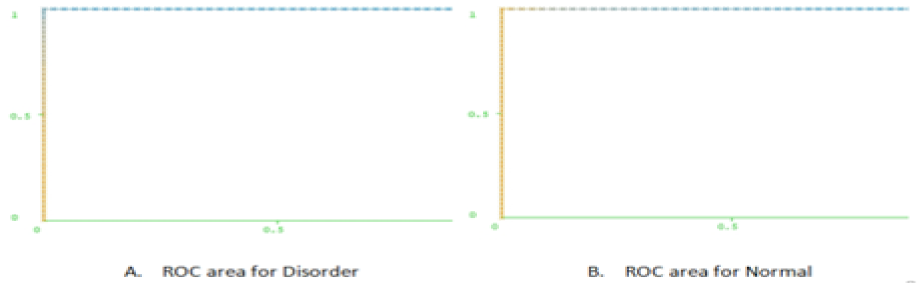


Fig-11. ROC Area under the curve for user supplied test set for single hidden layer ANN design. The threshold value for 90 instances of user supplied test set visualized using Weka to see the accuracy of the ANN design. x-axis: False positive rate and y-axis: True positive rate

4.2. Performance Evaluation Using Two Hidden Layers

The network parameters are set for two hidden layer design and the evaluation performance on the test set is calculated. The results show that 29 instances of the test split dataset are correctly classified with an accuracy of %93.5484, and two instances of the test split dataset are incorrectly classified with a mean squared error (MSE) value of 0.3485. The ROC area that visualizes the threshold curve of each Disorder and Normal class is shown in Fig. 12.

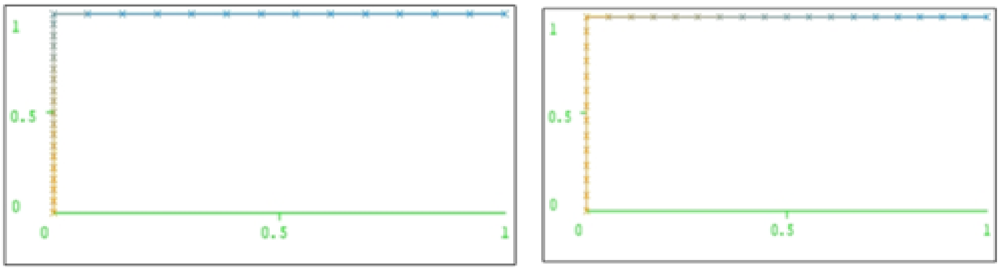


Fig-12. ROC Area under the curve of percentage split test for two hidden layer ANN design. The threshold value for 30 instances of the split test set for the training dataset Visualized using Weka, x-axis: False positive rate and y-axis: True positive rate.

To test the accuracy of the saved ANN design, a validation test sample is run based on the saved design and the performance evaluation is calculated. The result shows that 84 instances out of 90 instances of the user supplied test dataset are correctly classified with an accuracy of %93.333 %, and three instances of the supplied test dataset are incorrectly classified with mean squared error (MSE) value of 0.229. The ROC area that visualizes the threshold curve of disorder and normal class is shown in Fig. 13.

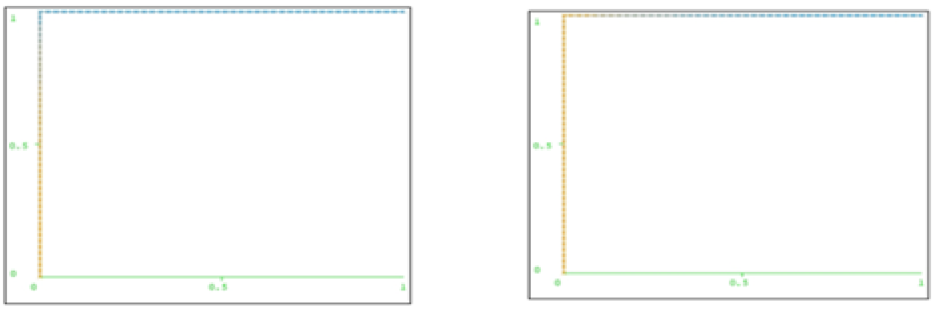


Fig-13. ROC Area under the curve for user supplied test set for two hidden layer ANN design. The threshold value of 90 instances for the user supplied test set visualized using Weka x-axis: False positive rate and y-axis: True positive rate.

4.3. Overall performance

The summary of the classification performance using the back-propagation neural network is given in Table-3. The single-hidden layer design of the network with ten neurons achieves the best classification rate and requires less processing time. The single-hidden-layer design achieves a classification performance of 96.774% for the test split dataset and 96.6667% for the user defined test dataset with MSE value of 0.1969 and true positive rate of 0.967.

Table-3. The experimental and performance comparison of the network design (collected by authors: shows the properties of each design with result accuracy for both training sample and user supplied test set)

Network Design	Training Time	Hidden Neurons	Time elapsed	Split Test Result	User supplied test result
1 Hidden Layer	15	10	0.1	96.774	96.6667
2 hidden layer	35	10, 10	0.1	93.5484	93.333

The proposed method is compared with others similar works as given in Table 4. The results expressively show the superiority of the proposed design. The proposed ANN performs its classification of IRI values to output the condition of the road as normal or disorder. The proposed single-hidden layer backpropagation ANN architecture outperformed with an accuracy of 97%.

5. CONCLUSION

In this work, an ANN with supervised learning architecture is designed to detect and identify road disorders. For the successful recognition of such hazards, the ANN training dataset is created by driving through designated roads with known anomalies. The vibrations arise from driving over such disorder, and normal road areas are recorded based on the International Roughness Index (IRI). The recording was carried out using an Android smartphone application that utilizes the device's built-in sensors, namely the accelerometers. The collected IRI-based data is applied through rigorous pre-processing operations to filter the collected data. The patterns are then extracted and normalized for the most effective features for the supervised training of the ANN.

The trained ANN successfully classifies different road conditions with a high success rate. The performance of the designed network is also compared with similar research work, whereby it shows the superiority of the results obtained. The approach used in this work and proposed method are considered to be cost-effective in comparison to other traditional methods that involve using proprietary equipment and vehicles to collect such road data.

Future work should consider the proper design of the cloud-service system to automatically train the artificial neural network (based on IRI values). Furthermore, some other evolutionary training algorithms can be integrated into the supervised learning part to train the network [24, 25] which would overcome some known shortages of the backpropagation training method. Moreover, the smartphone application could incorporate built-in pre-processing capabilities and the ability to connect to a cloud service that would help the municipals to take proper action to repair roads suffering from hazards as soon as possible.

Table-4. Comparison with related research work

Researcher	Data collection	Methods	Classification rate
Kattan and Aboalmaaly [1]	Smartphone application (No road standard measurements)	ANN	61%
Eriksson [4]	Attached sensor on the dashboard of the car (No road standard measurements)	Private machine learning algorithm (P2)	90%
Bhoraskar, et al. [17]	Smartphone sensors (No road standard measurements)	K-mean clustering and Support Vector Machine (SVM).	90%
Proposed Method	RoadBump pro smartphone application (Road standard measurements IRI)	ANN-backpropagation algorithm	97%

Funding: This study received no specific financial support.

Competing Interests: The authors declare that they have no competing interests.

Contributors/Acknowledgement: All authors contributed equally to the conception and design of the study. Our sincere thanks go to Grimmer Software especially Mr. David Grimmer for letting us to use their, RoadBumpPro ©, smartphone application freely.

REFERENCES

[1] A. Kattan and M. Aboalmaaly, "A smartphone-cloud application as an aid for street safety inventory," in *Proceedigns of the 11th International Conference on Electronics, Computer & Computation (ICECCO / IEEE)*, Abuja, Nigeria, 2014, pp. 6-7.

[2] G. Chugh, D. Bansal, and S. Sofat, "Road condition detection using smartphone sensors : A survey," *International Journal of Electronic and Electrical Engineering*, vol. 7, pp. 595–601, 2014.

[3] B. Hull, "Cartel: A distributed mobile sensor computing system," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*, Boulder, Colorado, USA, 2006, pp. 125-138.

[4] J. Eriksson, "The pothole patrol: Using a mobile sensor network for road surface monitoring," in *Proceedings of the 6th International Conference on Mobile Systems, Applications and Services (MobiSys 2008)*, Breckenridge, Colorado, USA, 2008, pp. 29–39.

[5] K. Chen, "Road condition monitoring using on-board three-axis accelerometer and GPS sensor," in *6th International ICST Conference on Communications and Networking in China (CHINACOM)*, Harbin, China, 2011.

[6] M. K. Hasan, N. Ahmed, and A. H. M. S. Islam, "Android mobile application: Remote monitoring of blood pressure," in *15th International Conference on Computer and Information Technology (ICCIT), University of Chittagong*, Bangladesh, 2012.

[7] K. Rachuri, "Smartphones based social sensing: Adaptive sampling, sensing and computation offloading," Ph.D Thesis, University of Cambridge, UK, 2012.

[8] N. Kalra and D. Bansal, "Analyzing driver behavior using smartphone sensors : A survey," *International Journal of Electronic and Electrical Engineering*, vol. 7, pp. 697–702, 2014.

[9] F. Orhan and E. P. Erhan, "Road hazard detection and sharing with multimodal sensor analysis on smartphones," in *7th International Conference on Next Generation Mobile Apps, Services and Technologies (NGMAST)*, Prague, Czech Repulbic, 2013, pp. 56-61.

[10] M. K. Hasan, "Road structure analysis using GPS information," in *Proceedings of the International Conference on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh*, 2013.

[11] B. J. D. Sutter, "Street bump app detects potholes, tells city officials, white paper, official website of the city of Boston." Retrieved from http://www.cityofboston.gov/news/Default.aspx?id=5490 2012.

[12] J. Whipple, W. Arensman, and M. S. Boler, "A public safety application of GPS-enabled smartphones and the android operating system," in *Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics (SMC'09)*, San Antonio, USA, 2009, pp. 2059-2061.

[13] Pavement Interactive, "Roughness, pavementinteractive.org. Available: http://www.pavementinteractive.org/article/roughness/," 2007.

[14] Minnesota Department of Transportation, "Rating overview." Retrieved from http://www.dot.state.mn.us/materials/pvmtmgmtdocs/Rating_Overview_State.pdf, 2003.

[15] A. Mednis, "Real time pothole detection using android smartphones with accelerometers," in *Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS)*, Barcelona, Spain, 2011, pp. 1-6.

[16] P. Mohan, R. Ramjee, and V. N. Padmanabhan, "Nericell: Rich monitoring of road and traffic conditions using mobile smartphones," in *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems (SenSys '08)*, Raleigh, NC, USA, 2008, pp. 323-336.

[17] R. Bhoraskar, N. Vankadhara, B. Raman, and P. Kulkarni, "Wolverine: Traffic and road condition estimation using smartphone sensors," in *Proceedings of the 4th International Conference on Communication Systems and Networks (COMSNETS 2012)*, Bangalore, India, 2012, pp. 1-6

[18] M. Cilimkovic, "Neural networks and back propagation algorithm," presented at the White Paper, Institute of Technology Blanchardstown, Dublin, Ireland, 2015.

[19] Y. M. Salih, "Detection of motorway disorders by processing of smart phone signals using artificial neural networks," Fatih University, M.Sc. Thesis, Department of Comptuer Engineering, 2016.

[20] I. O. Dataschool, "Simple guide to confusion matrix terminology." Retrieved from http://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/," 2014.

[21] Grimmersoftware, "Road bump pro." Retrieved from http://www.grimmersoftware.com/roadbump.html, 2015.

[22] Grimmersoftware, "Road bump user's guide." Retrieved from http://nebula.wsimg.com/5d5f829c0bc5d7b768a5c2769c424535?AccessKeyId=77FD3AE629701C2E9990&dispositio=0&alloworigin=1, 2015.

[23] M. A. Hall, I. H. Witten, and E. Frank, *Data mining: Practical machine learning tools and techniques*, 3rd ed.: Morgan Kaufmann Publishers, 2011.

[24] A. Kattan, R. Abdullah, and R. A. Salam, "Harmony search based supervised training of artificial neural networks," in *Proceedings of the 1st International Conference on Intelligent Systems, Modelling and Simulation (ISMS2010)*, 2010, pp. 105-110.

[25] A. Kattan and R. Abdullah, "Training feed-forward artificial neural networks for pattern-classification using the harmony search algorithm," in *Proceedings of the 2nd International Conference on Digital Enterprise and Information Systems (DEIS2013)*, 2013, pp. 84-97.