DOI: 10.48011/asba.v2i1.1274

Vibration Detection of Vehicle Impact Using Smartphone Accelerometer Data and Long-Short Term Memory Neural Network

João Paulo Brognoni Casati [*]
Ruy Alberto Corrêa Altafim [*, **]
Ruy Alberto Pisani Altafim [**]

[*] *Electrical and Computer Engineering Department, São Carlos Engineering School , University of São Paulo, SP - Brazil (e-mail: jpcasati@usp.br)*
[**] *Computer Systems Department, Informatics Center, Federal University of Paraíba, PB - Brazil*

Abstract:
Vehicle's analysis can be useful for a variety of traffic problems, such as monitoring road damages and vehicle type classification. Further, traffic behavior analysis can be useful to monitor traffic jams as a smart cities solution. In this paper, the vibration caused by a vehicle passing through a speed bump was recorded with a docked smartphone. The acquired signals were processed in order to detect the generated impact. In order to analyze this data a LSTM neural network was used due to its classification process over time while the smartphone accelerometer was continuously operating (waiting for a vehicle pass by). This deep learning technique allows the use of raw 3-axis accelerometer data. The results achieved 98% of accuracy with a low level of false positives (less than 1%). Indicating that the methodology is effective in classification of vehicles by their impact vibration.

Keywords:
artificial neural networks; deep learning; Long-Short Term Memory; accelerometer; traffic analysis; smart cities.

1. INTRODUCTION

The use of accelerometer signals to represent real world activities is widespread in scientific area. There are many application in various fields, some of them are:

- Medicine, as Parkinsonian Tremor (Papadopoulos et al., 2019);
- Gesture identification (Josephs et al., 2020);
- Smart Cities (Habibzadeh et al., 2017);
- Vehicle detection and classification (Obertov and Andrievsky, 2014), (Rivas et al., 2017);
- Road condition analysis (Gueta and Sato, 2017), (Bello Salau et al., 2019).

A particular interesting application developed by Bales et al. (2016) is the gender classification using accelerometer signals. This approach includes high-sensitivity accelerometer beneath the floor of a building and acquired the vibration signals. For the data analysis, they applied Fast Fourier Transform (FFT) in the samples and then used three machine-learning methods for the classification: Boosted Decision Trees; Support Vector Machines (SVM); and Artificial neural Networks. The data was processed using FFT in this case, because shallow learning algorithms do not work well using some types of raw data (LeCun et al., 2015). Although, the authors achieved 80% accuracy using Boosted Decision Trees algorithm.

Besides gender classification, there are a great amount of studies concerning terrain and road condition classification. Some of these methods are based on image processing approach (Danti et al., 2012). However, this approach is computationally costly and it has other concerns such as privacy and the expensive equipment needed (Gunawan et al., 2015)

A more feasible and efficient method for road monitoring, make uses of accelerometer to provide vibrating signals. And they can be used to classify the terrain using a planetary exploratory rover, such as those developed by Brooks and Iagnemma (2005), where the authors equipped a rover robot with an accelerometer and acquire the signals to process and classify the type of terrain. The accuracy results reported were high, with a highlight for sand terrain, which authors concluded that the confidence of for this method was 100%.

As pointed by Obertov and Andrievsky (2014), vehicle classification and detection can provide quality information for a variety of fields, such as accident handling, traffic measurements and control, statistics of road traffic and condition, road damage prediction, and others. In this paper, a data acquisition system is created, in order to detect and save the vibration impact of vehicles passing through a speed bump. The impact raw data from an accelerometer, was saved, processed and classified differentiating the vibrations in two classes (with vehicle impact and without vehicle impact). The analysis and validation of data were made using deep neural networks architecture, which can provide robust models for raw data processing and classification (LeCun et al., 2015).

The chosen architecture was the Long-Short Term Memory (LSTM), which can provide a way to analyze temporal series and classify them along time (Kang, 2017).

For a better understand the paper was dived as follows: in section 2 is presented the development of a data acquisition system, which provides a way to save vehicle vibration data. In Section 3 the acquired data was validated using LSTM neural network. Further, in section 4 some of the results of data analysis are shown, in order to validate the acquired data. Finally, in section 5 some conclusions and future work concerning the application development are provided.

2. DATA ACQUISITION

The aim of this work is to promote a method to classify vehicles according to their vibration signals produced by them when they are passing in a traffic road. Therefore, in order to create a vibrating signature by the vehicles, a handmade speed bump was constructed with a docked smartphone accelerometer attached to it. An embedded application was developed to continuously record the vibrations. These signals were later used in the data processing.

2.1 Speed Bump

The speed bump was constructed, mainly in wood, to promote a vehicles vibration as they passes through it. The picture presented in Figure 1 shows an actual photography of the speed bump.

Figure 1. Actual photo of the handmade developed speed bump.

Wood was chosen as basic material due to their strong characteristic i.e. to withstand a vehicle going over, they ease to be processed and are lightweight for transportation. To ensure vibrating conditions gaps between the tacos beneath the main wooden plate were implemented, as can be accurately seen in Figure 2. In total 11 tacos separated by gaps and fills were implemented in the speed bump. Additional information about the wooden speed bump is provided in table 1.

In order to record vehicle traffic, the speed bump was placed in residential street and, as a vehicle passes over, the vibrations captured by the accelerometer were recorded in the smartphone.

Figure 2. Zoomed photo for the taco detail in the speed bump.

Table 1. Speed Bump Measures in centimeters.

Measure	Size (cm)
Principal Plate Length	283
Principal Plate Width	18.8
Principal Plate Height	1.5
Taco Length	18.8
Taco Width	3
Taco Height	1.5
Gap Size	25

The speed bump was painted yellow and positioned in a way that passes cars and motorcycles, presented in Figure 3.

Figure 3. The handmade speed bump with the smartphone docked and positioned to acquire real data.

2.2 Acquiring and Saving Data

The smartphone employed in the data acquisition was a Motorola G6 Plus running Android operational system. There are several methods to record the data prevenient from a smartphone accelerometer, however to save the data in a proper way for future analysis, a smartphone application was implemented.

The software named here KaptAccel is now available for free at Google Play and by scanning the image in Figure 4 one may download the app. An image of the main interface of the developed mobile application is shown in Figure 5.

The mobile application was developed for Android using Kotlin programming language. It is useful for acquiring 3-axis data of smartphone accelerometer, saving it in a CSV file with 5 columns.

Each column of the saved file has one type of data. The first has timestamp (got from the mobile device time), second has the X-axis of the accelerometer, the third column has the Y-axis and the fourth has the Z-axis. There is a fifth column which has the number of the class marked in the

Figure 4. QR Code for access to the KaptAccel mobile application.

app, in order to facilitate the acquisition of labeled data aiming pattern recognition.

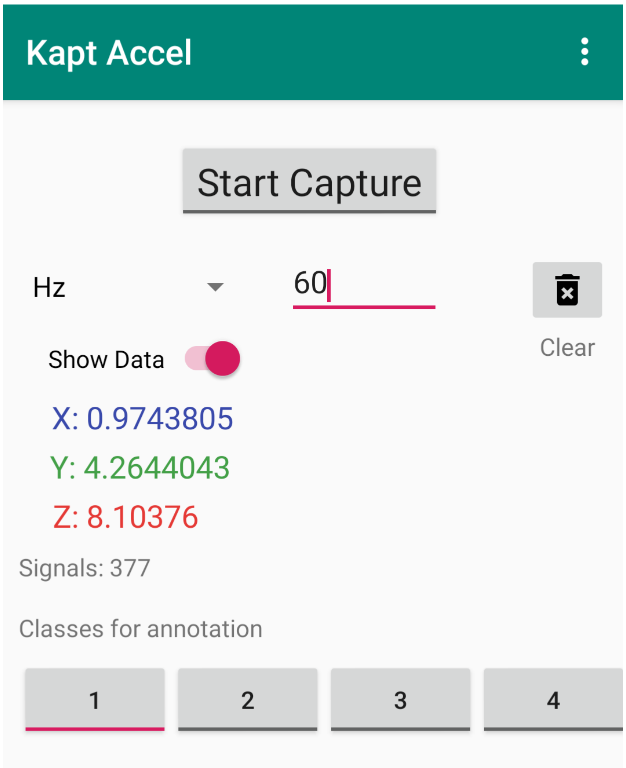

Figure 5. The main screen of KaptAccel in use.

The mobile application was developed for Android using Kotlin programming language. It is useful to acquiring 3-axis data of smartphone accelerometer, saving the data in a CSV file with 5 columns.

Each column provides one type of information. The first column represents the timestamp (obtained from the mobile device time). The second, third and fourth columns show the X-axis, Y-axis and Z-axis of the accelerometer, respectively. Finally, in the fifth column, a number of the class marked in the app, is provided to facilitate the acquisition of labeled data aiming pattern recognition.

The data acquisition frequency employed here, is native from the Android operating system, which varies during the process as axis values changes. Nevertheless, a rate between 20 and 25 3-axis signals per second was recorded leading a total of 39,060 3-axis vibration samples.

In this initial approach the difference between types of vehicles is not verified, therefore only two classes are represented:

- 0: Without vehicle vibration; and
- 1: With vehicle vibration.

3. DATA ANALYSIS

The data provided by the system (speed bump and smartphone with KaptAccel) has 3 dimensions inputs and 1 dimensional output, which can represent the two classes (zero for negative vehicle impact vibration, or no vibration, and one for positive impact vibration, or a vehicle is passing through the speed bump).

An example plot of the data is shown in Figure 6. In this chart, the area without vibration is marked as class 1, where the other area is marked as class 0. It creates a step function, which is used to train the neural network and then get metrics of its accuracy.

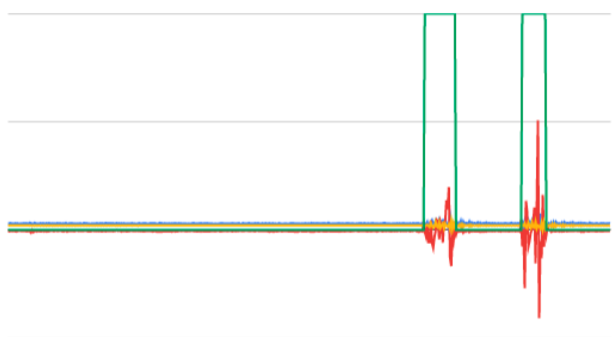

Figure 6. Chart of 500 labeled samples of the collect data.

In order to analyze data through time, the LSTM neural network architecture was chosen because of its capability of memorize events for long time and classify time series on the go (Kang, 2017). Another advantage for the use of LSTM is that it works fine with raw data examples, as done for the major deep learning architectures (Brownlee, 2018). Further, there are several libraries that implement LSTM architecture (Keras, 2020).

The configuration used for the layer in this model has 32 LSTM cells, which can memorize the amount of data and its variations. The output data is a Dense layer with 1 neuron, which can differ between the 2 classes. As the input value varies along time, there are some batches of it that can produce one output. The tests were made using 10, 20, 30, 40 and 50 samples in the input, for the output. This means that, when using 10 entries for the analysis, less than 0.5 second is used to verify if there is a vehicle passing through the speed bump. Or when using 50 samples, almost 2 seconds are needed to verify it.

The amount of samples (39,060) was randomly divided as: 70% for training set and 30% for test set. The neural network has been trained with each size of input for 100 epochs. The best loss function used to verify it with training weights was the Mean Squared Error.

To pursuit better results, training weights that provided the minor value of loss was saved and used as test set. After training the LSTM using 70% of the entire data set, the other 30% was saved for future analysis.

The prediction of the test set generates some data and it was analyzed considering it as right or wrong when the trained LSTM defined a class.

4. RESULTS AND DISCUSSION

In this section the results of the previously analysis are presented. The first result generated is the value of loss function when training the LSTM with the 70% of the data set destined for training set. As explained in Section 3, it was performed some training, varying in the size of input data for each time step. In Table 2 are presented the results of the Mean Squared Error (MSE) for these trainings after 100 epochs.

Table 2. Results of the training Mean Squared Error (MSE).

Input feature	MSE value
10 samples	0.01206
20 samples	0.01421
30 samples	0.00820
40 samples	0.00824
50 samples	0.00901

As one can observe in Table 2, there are some differences between the trainings. After increase the number of samples in the input, the MSE tends to decrease, but, after some increasing in the input size, this trend vanishes. A comparison of the values is shown in Figure 7. This chart shows that the training that achieved the best performance was the one using inputs of 30 samples each, followed by the training with 40 input samples.

After the training step, LSTM uses test data to predict the classes. For each of the input size, the prediction performed presents a high accuracy.

Table 3, indicates the results of true positives, true negatives, false positives, false negatives and accuracy for the test set with inputs that has 10 samples each. It is possible to observe that the low value of false positives indicates that the process of prediction difficultly considered some amount of vibration as a vehicle passing through the speed bump.

Table 3. Results with input of 10 samples (%).

Measure	Percentage
True Positives	5.26
True Negatives	92.76
False Positives	0.61
False Negatives	1.36
Accuracy	98.03

In Table 4 the prediction results considering inputs of 20 samples each are presented. In this case, the false positives

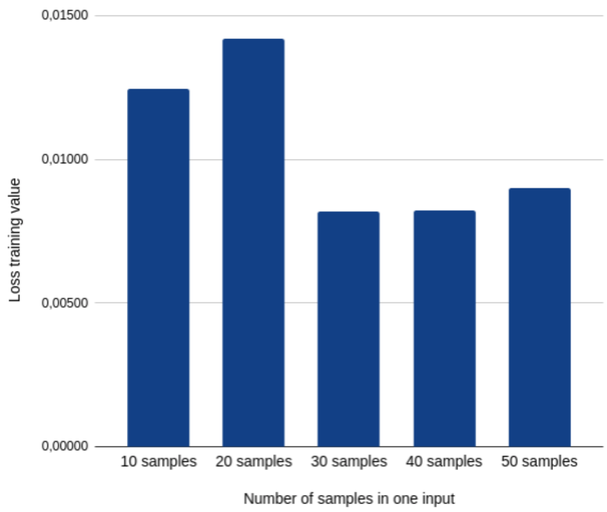

Figure 7. Comparison of training loss value versus number of samples in each input.

is slightly higher than the prediction with inputs of 10 samples, and the accuracy value is also decreased. Nevertheless, the result values are very close, but a comparison between this to inputs lead to a worse case scenario for an input with 20 samples.

Table 4. Results with input of 20 samples (%).

Measure	Percentage
True Positives	5.61
True Negatives	92.23
False Positives	0.80
False Negatives	1.35
Accuracy	97.85

Considering the inputs with 30 samples, a lower loss value is obtained in the training step. These results shows an improved performance, reducing false positives and false negatives, increasing the accuracy to 98.59% as presented in Table 5.

Table 5. Results with input of 30 samples (%).

Measure	Percentage
True Positives	6.00
True Negatives	92.60
False Positives	0.53
False Negatives	0.88
Accuracy	98.59

In Table 6 are presented the results of test set prediction using 40 samples per input. Considering loss value in the training step, it is expected that in this case the accuracy is to be maintained. With an accuracy of 98.76%, it shows that this prediction performed well, being the highest of the experiment. Another point of highlight is the low value of false positives, only 0.35%. It means that is rare to consider something as a vehicle vibration, when it is not.

Nevertheless, in the last test performed with 50 samples for each input (Table 7), the achieved results were very close to the other tests. The high Accuracy and low false positive value shows that it can be used to detect vehicle impact events. The problem is that similar or better accuracy

Table 6. Results with input of 40 samples (%).

Measure	Percentage
True Positives	6.05
True Negatives	92.71
False Positives	0.35
False Negatives	0.90
Accuracy	98.76

can be achieved using smaller inputs, which has lower computational cost.

Table 7. Results with input of 50 samples (%).

Measure	Percentage
True Positives	6.03
True Negatives	92.57
False Positives	0.57
False Negatives	0.82
Accuracy	98.61

In Figure 8 is presented a performance comparison between the 5 tests (5 different input sizes), considering the accuracy. And as observed, there results are similar, but the input sizes of 30 and 40 presented a better performance.

Further, if computational cost is consider, which is not the object in analysis of this paper, it is acceptable that the input size of 30 had better performance overall.

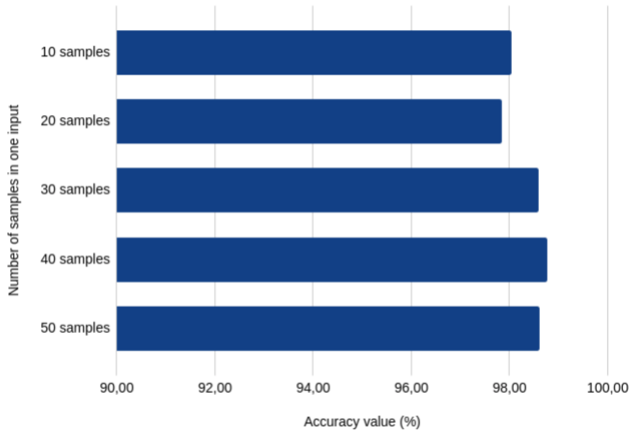

Figure 8. Comparison of accuracy value in percentage considering the test set of samples.

5. CONCLUSIONS AND FUTURE WORK

The intelligent analysis presented in this paper states that accelerometer data can be used to detect impact of vehicles in a speed bump placed on the street. Some contributions of this paper are: The development of traffic data acquisition system; Road damage analysis and prediction; Traffic analysis and prediction; Primary application (add more classes needed) for vehicle type classification; Strength of the impact caused by vehicles in irregular surfaces. Some other steps is needed to show more of the potential of artificial intelligence applied to this kind of problem.

This paper can be extended for future work. Some of the points that are being developed: The amount of collected data has three types of vehicles and one of the next steps is to develop a robust classifier to differentiate it. The extension of sample data, making more experimental sessions; Intelligent extension of sample data using Generative Adversarial Networks (Goodfellow et al., 2014); Increase the number of classes, classifying different types of vehicles; Analyze data obtained by the other accelerometers; The use of other deep learning architectures as Convolutional Neural Networks and Autoencoders in order create more robust intelligent models for processing great amounts of raw data (Goodfellow et al., 2016); Using deep learning architecture, raw data and achieving high values of accuracy, this work shows potential to be extended and useful for many other works in the area.

Nevertheless, the focus of this paper was to develop a data acquisition system and validate the acquired signals using artificial neural networks.

REFERENCES

Bales, D., Tarazaga, P.A., Kasarda, M., Batra, D., Woolard, A.G., Poston, J.D., and Malladi, V.V.N.S. (2016). Gender classification of walkers via underfloor accelerometer measurements. *IEEE Internet of Things Journal*, 3(6), 1259–1266.

Bello Salau, H., Onumanyi, A., Aibinu, A., Onwuka, L., Dukiya, J., and Ohize, H. (2019). A survey of accelerometer-based techniques for road anomalies detection and characterization. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE AND APPLICATION*, 3, 8 – 20.

Brooks, C.A. and Iagnemma, K. (2005). Vibration-based terrain classification for planetary exploration rovers. *IEEE Transactions on Robotics*, 21(6), 1185–1191.

Brownlee, J. (2018). How to develop rnn models for human activity recognition time series classification. URL `https://machinelearningmastery.com/how-to-develop-rnn-models-for-human-activity-recognit` [Online; accessed 22 April 2020].

Danti, A., Kulkarni, J., and Hiremath, P. (2012). An image processing approach to detect lanes, pot holes and recognize road signs in indian roads. *International Journal of Modeling and Optimization*, 2, 658–662. doi: 10.7763/IJMO.2012.V2.204.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks.

Gueta, L.B. and Sato, A. (2017). Classifying road surface conditions using vibration signals. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 039–043.

Gunawan, F.E., Yanfi, and Soewito, B. (2015). A vibratory-based method for road damage classification. In *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 1–4.

Habibzadeh, H., Qin, Z., Soyata, T., and Kantarci, B. (2017). Large-scale distributed dedicated- and non-dedicated smart city sensing systems. *IEEE Sensors Journal*, 17(23), 7649–7658.

Josephs, D., Drake, C., Heroy, A., and Santerre, J. (2020). semg gesture recognition with a simple model of attention.

Kang, E. (2017). Long short-term memory (lstm): Concept. URL `https://medium.com/@kangeugine/long-short-term-memory-lstm-concept-cb3283934359`. [Online; accessed 19 April 2020].

Keras (2020). Keras documentation. `https://keras.io/`. [Online; accessed 21 May 2020].

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. doi:10.1038/nature14539. URL `https://doi.org/10.1038/nature14539`.

Obertov, D. and Andrievsky, B. (2014). Vehicle classification using measurements from accelerometers mounted on the road surface. In *2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 413–417.

Papadopoulos, A., Kyritsis, K., Klingelhoefer, L., Bostanjopoulou, S., Chaudhuri, K.R., and Delopoulos, A. (2019). Detecting parkinsonian tremor from imu data collected in-the-wild using deep multiple-instance learning. *IEEE Journal of Biomedical and Health Informatics*, 1–1. doi:10.1109/jbhi.2019.2961748. URL `http://dx.doi.org/10.1109/JBHI.2019.2961748`.

Rivas, J., Wunderlich, R., and Heinen, S.J. (2017). Road vibrations as a source to detect the presence and speed of vehicles. *IEEE Sensors Journal*, 17(2), 377–385.