GEOSCIENCES

An evaluation of machine learning methods for speed-bump detection on a GoPro dataset

JOHNY MARQUES, RAULCEZAR ALVES, HENRIQUE C. OLIVEIRA, MARCO MENDONÇA & JEFFERSON R. SOUZA

Abstract: Every day, new applications arise relying on the use of high-resolution road maps in both academic and industrial environments. Autonomous vehicles rely on digital maps to navigate when optical sensors cannot be trusted, such as heavy rainfalls, snowy conditions, fog, and other situations. These situations increase the risks of accidents and disable the potentials of real-time mapping sensors. To tackle those problems, we present a methodology to automatically map anomalies on the road, namely speed bumps in this study, using an off-the-shelf camera (GoPro) and Machine Learning (ML) algorithms. We acquired data over a series of differently shaped speed bumps and applied three classification techniques: Naive Bayes, Multi-Layer Perceptron, and Random Forest (RF). With over 96% of classification accuracy, then RF was able to identify speed bumps on a GoPro dataset automatically. The results show a potential of the proposed methodology to be developed in surveying vehicles to produce highly-detailed maps of vertical road anomalies with a fast and accurate update rate.

Key words: Feature classification, mobile mapping, machine learning, pattern recognition, speed bump, urban mapping.

INTRODUCTION

According to the 2018 Revision of World Urbanization Prospects delivered by the United Nations, the majority of the world population (approximately 55%) lives in urban areas. However, this percentage is even higher in some parts of the globe, such as Northern America (82%), Latin America (81%), Europe (74%) and Oceania (68%). Because road systems are an essential part of urban mobility projects, governments spend a considerable amount of resources to create and especially maintain the road system as cities grow.

The road network must have its structure well surveyed, due to its importance for traffic management. These documents are crucial for city planning, road inventory, safe driving, and

correlated applications (Aljaafreh et al. 2017, He & Orvets 2000, Wang et al. 2016 and Zhao et al. 2011). The most important features to be mapped in a road inventory are the central road axis, traffic signs, potholes, and speed bumps. As presented by (Aljaafreh et al. 2017), road network monitoring and maintenance is crucial to increase the safety of drivers and pedestrians.

As with any mapping activity, the data acquisition portion of this task is more expensive (He & Orvets 2000). Data are obtained by the land survey, aerial photogrammetric missions, orbital imagery, and mobile mapping systems. These platforms are chosen according to the project, taking into consideration the desired scale of the output. Nowadays, Mobile Mapping Systems (MMS) are used for road network mapping (Tao & Li 2007 and Molina et

al. 2016). These systems are composed by a set of sensors such as cameras, Global Navigation Satellite Systems (GNSS) receivers/antennas, inertial measurement units, and Light Detection And Ranging (LiDAR) systems. Recently, MMS are assembled in platforms with low-cost sensors using GoPro cameras. The GoPro includes a GNSS receiver, gyroscopes, accelerometers, image sensor, and a thermometer.

The dataset acquired during an MMS mission is usually large and requires a long time of post-processing, with a considerable amount of human intervention. Several studies have been done to improve the final product quality and decrease production time. Most of the research in the field uses computational applications for improving the aspects of map production.

The contribution of this research is the evaluation of speed bump detection applying Machine Learning (ML) techniques using a dataset from a GoPro camera by a low-cost mobile mapping system.

The remainder of this paper is organized as follows: Related Work presents and discusses the related work to speed bump and traffic features detection. Also, the proposed methodology is presented in Materials and Methods. The experimental setup, results and analysis are shown in Results. Lastly, Conclusions concludes and suggests directions for future work.

RELATED WORK

There are several types of data and sensors used to solved the mentioned problems in the introduction, such as accelerometer measurements, satellite positioning, gyroscopes, LiDAR, and images. The methods used range from fuzzy inference, unsupervised learning, and genetic algorithms.

In Bello-Salau et al. (2018), the authors present an algorithm for detecting potholes and bumps from acceleration signals using an accelerometer. A wavelet filter was applied to decompose the signals in scales. Then, road anomalies were identified on a threshold system using features extracted from the wavelet coefficients. The system detects road anomalies with high levels of accuracy, precision, and low false alarm rates.

The work of Soilán et al. (2018) describes a methodology for the extraction of semantic information from public data aiming at urban parameterization. First, a building is segmented with voxel data. After, heuristic rules and unsupervised learning were applied to the ground surface data to distinguish the sidewalk and pavement points. Finally, the system was able to generate an F-score close to 95% for detecting pavement and path.

Zhao et al. (2011) proposes the extraction of urban roads from airborne LiDAR data. The methodology consists of the terrain separation from Digital Surface Model (DSM), road centerline extraction, and verification of road networks. The elevated objects are removed from the depth image. After, missing roads, the inference is executed on-road centerline vector map according to gestalt laws. A direction-based voting technique is developed to evaluate the reliability of each path. Results show that the road features are successfully projected onto the depth image.

In Wang et al. (2016), the authors analyze different road extraction methods. Firstly, the road features extraction were investigated. Secondly, the authors present the advantages and disadvantages of various ways. Then, to improve the classification results, the road extraction combines multiple methods according to the real applications. Results show that the methodology could successfully recognize roads

using different road features, but when objects like water, buildings, trees, grass, and cars occlude the road.

Aljaafreh et al. (2017) propose a speed bump detection method based on a fuzzy inference system (FIS). The FIS detects the speed bumps from vertical acceleration and the speed of the vehicle. The system uses the accelerometer sensor in a smartphone. The methodology is evaluated at different speed levels. Results show that the system is promising for bumps detection.

Varma et al. (2018) proposes an image-based speed hump and bump detection method. The technique uses deep learning and stereo cameras to detect speed humps and bump, which make distance calculations for detected objects. Results show that the technique obtained has an accuracy of 97.44% for tagged objects and 93.83% for unlabeled objects.

Devapriya et al. (2015) proposes a methodology that uses images. The captured video is transformed into an image and then pre-processed on that image. After that, morphological operations are performed and the projection of the object in the image. The methodology was based on five different types of speed bumps, for type 1 the accuracy was 90%, type 2 in 85%, type 3 in 83%, type 4 in 80% and type 5 at 4%.

Fernández et al. (2012) it has a detection system of free space and spines, these spines being defined by the Spanish government. Free space detection is performed using a low-cost handle, and spine detection are performed with images captured by a camera, taking into account the zebra-like painting that is done on these spines. The image is processed to work only with the regions of interest that go through an edge and contour extraction phase, and then the lines in the image are analyzed. Accuracy

was 100% for spines and 94% for detection of zebra bands.

Considering the related work presented, it is possible to affirm that the proposed methodology is original, due to the fact that none of the ML algorithms used in this work were already applied for identifying this type of features - speed bumps and also only one of these related work uses data of gyroscope, accelerometer, and GNSS, even though with a significantly different methodology.

MATERIALS AND METHODS

The road centerline mapping is an essential task for a land traffic control. This task makes informed decisions since it facilitates finding their anomalies, then allowing agility in the process of repairing and managing areas that require speed reduction.

A centerline map has the potential to be integrated with multi-sensor navigation route applications, so drivers and autonomous systems can be aware of upcoming road anomalies. This warning can assist drivers in situations where a lack of visibility due to rain or snow, and lack of illumination are presented as well as the presence of wrong or outdated traffic signals.

In this context, this paper proposes a methodology that is organized in 4 main steps, as presented in Fig. 1 for an automatic speed bump mapping using data from a GoPro device in a vehicle, with a post-processed ML-based algorithm. The purpose of this article is to present a method to detect coordinates of road locations where anomalies are present and classifying them into two main categories:

- 1) vertically low and horizontally long platforms
- 2) vertically high and horizontally short bumps

Figure 1. Flowchart of proposed methodology.

For each categories, the driver's action may vary.

The main idea of the proposed methodology is to evaluate different ML algorithms to extract geographic coordinates that represent speed bumps. The steps presented in Fig. 1

can be described as follows: a low-cost set of cameras is assembled in a vehicle which is driven along an specific trajectory. The camera that contains a set of sensors is recording data during this trajectory. As soon as the vehicle approaches a speed bump, the driver tends to slow down the vehicle (varying the acceleration in the x-axis). During the movement passing by the speed bump the vehicle is going up and down (varying the acceleration in the z-axis), and to complete the movements observed by the sensors, the vehicle front part will point up and down (varying the gyro in the y-axis). Therefore, these observations, followed by a training dataset, and a ML algorithm, may allow the mapping of speed bump in an automatically way.

Video capture

The information regarding the road centerline vertical profile is captured by a vehicle which is adapted with a device for assembling the

Figure 2. Vehicle and base for the GoPro cameras.

GoPro® cameras. Fig. 2 presents the vehicle, the platform and the mentioned device. In this work, only one of the cameras - the top one - is used. However, the same approach can be applied for all of them and a more precise result obtained due to the measurements redundancy.

Sensors data extraction

After capturing the video, a pre-processing step is performed to extract information such as acceleration, rotation, and coordinates that the camera generates as metadata in the videos. For this task, we used a Python script and the FFmpeg (2018) tool to extract the metadata from the video creating a ".bin" file. With this file, we call the "gopro-utils" library (2018), which from the ".bin" file as an input, generates four "csv" files containing the video three-axial acceleration and rotation, as well as three-dimensional coordinates and the respective timestamps.

Time synchronization

Since the sensors (gyroscope, GPS, camera, and accelerometer) collect data at different rates, then it is necessary to perform a synchronization between the acquired data to produce a single

file with coordinates, acceleration, and gyro from a specific moments.

The first part of the script converts the GPS time to local time. After this conversion, we make the interpolation between the file with rotation information (gyro), acceleration, and coordinates since they are collected at different rates. The interpolation used is a linear approach that provides the expected accuracy of the position, and it is based on the following equation:

$$y_i = y_1 + (y_2 - y_1) \times \frac{t_i - t_1}{t_2 - t_1} \quad (1)$$

y_i being the parameter value at the i th time with i between timestamps 1 and 2; y_n , the parameter value of observation n ; and t_n the timestamp of observation n .

With the synchronized data, it is possible to extract the signal features, such as the peak of a particular period, the time in which the vehicle raised when crossing a speed bump, the average acceleration, and average speed. Fig. 3 shows the samples types, which we are feeding into the training portion of the ML algorithms:

- 1) Type 1: long platforms.
- 2) Type 2: short bumps.
- 3) Type 3: small irregularities on the road or flat road that are neither Type 1 nor Type 2.

Although the purpose of this application is to detect long platforms and short bumps along the road, an ML classifier needs to have classes that could represent false positives. In this case, Type 3 represents readings that should

not be classified as Type 1 or 2. Then load the interpolated file as an array and iterate over that array. Each record that exceeds the threshold, as shown in Fig. 3, is placed in an auxiliary matrix. The parameters of interest are:

- Δt is the amount of time that the vehicle remained above the indicated threshold.
- $\overline{Ax}$ is the average acceleration of x during the period in which the vehicle remained above the indicated threshold.
- $\overline{Ay}$ is the average acceleration of y during the period in which the vehicle remained above the indicated threshold.
- $\overline{Az}$ is the average acceleration of z during the period in which the vehicle remained above the indicated threshold.
- $\overline{V}$ is the average speed during the period in which the vehicle remained above the indicated threshold.
- P is the highest value of rotation in the y -axis (transversely to the drive direction - which describe the camera's up and down movement) in the period in which the vehicle remained above the indicated threshold.

After synchronization between the original files, the threshold applied to the information. Thus, a dataset with the attributes presented above is generated, which will be used in training and later in the classification step. As discussed in the next session.

ML

The machine learning algorithms operate by constructing a model from sample entries to make predictions or decisions guided by the data rather than by merely following inflexible and static information. The application of ML in this work aims to build models capable of correctly detecting and classifying speed bumps on the road. Three different algorithms available on Weka (2018) tool were used: Naive Bayes

Figure 3. Types of samples.

(NB), Multi-Layer Perceptron (MLP) and Random Forest (RF).

Naive Bayes

It is a learning technique based on the Bayes theorem of conditional probability. Even though a simple model, this theorem is surprisingly effective when used as a classification algorithm (Julia 2018). Equation 2 shows the Bayes Theorem, used as a basis for this ML technique:

$$P(c|x) = \frac{P(x|c) \cdot P(c)}{P(x)} \quad (2)$$

In simple terms, and applying the explanation to this study, this classifier estimates what the probability of the collected data from the sensors to belong to data collected over a speed bump is. The posterior function is then maximized to classify a specific feature as defined in the training phase.

Multilayer Perceptron

MLP networks use at least one intermediate layer between input data and the features to be classified. Potential areas of application of this technique are (Silva et al. 2010): a universal approach of functions, pattern recognition, process identification and control, forecasting time series, and system optimization. In our work, MLP is evaluated under its pattern recognition capabilities on a multidimensional dataset.

Random Forest

It is a supervised learning algorithm that builds multiple decision trees and merges these trees to achieve an accurate prediction. Each decision tree passes through a voting mechanism, which chooses the most voted classification (Dantas 2015). In this study, we explore the classification capabilities of the RF method to, similarly to the MLP network, classify a multidimensional

dataset into features of interest. In this study, the “branches” of each tree will be defined by the information provided by each parameter used as an input (Fig. 3).

RESULTS

To assess the proposed methodology, a set of experiments were performed on the premises of the University of Campinas main campus. Three videos were used, two of them for training and another to test the classification models. After pre-processing the training videos, 7500 instances were generated. Most of these instances belong to Type 3, which is more common on a regular road. Therefore, to avoid overfitting on our classifiers, we first applied a technique to balance the dataset.

A Class Balancer filter was used to reweights the instances in the dataset so that each class has the same total weight. The total sum of weights across all instances is maintained. For the last video, used to test our models, the pre-processing step acquired 32 instances. In this case, they were classified without balancing. A video, that included all the gyroscope, accelerometer and

Figure 4. Route performed for the experiment.

positional parameters, as well as the required meta-data, was collected along a 5km route represented in Fig. 4.

For the ML process, we applied k-fold cross-validation with $k=10$, i.e., the original sample is randomly partitioned into ten subsamples. Of the ten subsamples, a single subsample is retained as the validation data for testing the model, and the remaining nine subsamples are used as training data. The cross-validation process is then repeated ten times (the folds), with each of the ten subsamples used exactly once as the validation data. The ten results from the folds can then be averaged (or otherwise combined) to produce an estimation.

Fig. 5 shows the accuracy of each classifier after the training phase. Naive Bayes reached 64.83% of accuracy, then followed by RF 63.23% and MLP 33.33%.

Once the models were trained, we used them to classify the data from the last video. This time, RF performed in a better way, as depicted by Fig. 6. It achieved 96.84% of accuracy with only 1 false positive. The second best was Naive Bayes with 13 correct classifications and 19 false positives, which corresponds to 40.62%. As in the training phase, the worst classifier was MLP summing 26 false positives against 6 corrected classification (18.70%).

Table I, Table II, and Table III present the confusion matrix of each classifier on test data. Rows and columns report the number of false positives, false negatives, true positives, and true negatives. This allows more detailed analysis than mere proportion of correct classifications (accuracy).

Fig. 7, Fig. 8 and Fig. 9 illustrate the classified instances along the map using each ML approach. Green markers show correct classifications while the red markers the wrong ones. The labels next to the markers represent the resulting classification, where the left number is the actual type of the

Figure 5. Training accuracy of the classifiers.

Figure 6. Test accuracy of the classifiers.

Figure 7. Map created with Naive Bayes.

instance, and the right number is the predicted type.

We observed based on the presented results, that the imperfections/anomalies along the streets affect the metrics used, as well as cases of lack of maintenance both on the pavement and on the speed bumps.

Figure 8. Map created with Random Forest.

Figure 9. Map created with MLP.

Table I. Confusion Matrix of Naive Bayes.

Types	Type 1	Type 2	Type 3
Type 1	5	1	0
Type 2	9	6	0
Type 3	0	9	2

Table II. Confusion Matrix of Random Forest.

Types	Type 1	Type 2	Type 3
Type 1	6	0	0
Type 2	0	15	0
Type 3	0	1	10

Table III. Confusion Matrix of MLP.

Types	Type 1	Type 2	Type 3
Type 1	6	0	0
Type 2	0	15	0
Type 3	0	1	10

The speed bumps in Brazil are regulated by the National Traffic Code Resolution nº 600/2016, but there are cases where these features do not fit to the technical requirements, which can even cause accidents, and in the problem tried to be solved in this work could lead to false positives.

In this work, as pointed out previously, only the data generated by one of the cameras were used, and the results can be refined using the data generated by the other four cameras, as well as applying deep learning to the images captured by the front cameras.

CONCLUSIONS

The digital maps used in autonomous vehicles need a collaborative high update rate. This method proves that using low-cost mapping tools, such as a base-mounted GoPro camera along with an ML technique, it is possible to achieve an accurate and efficient way for road map features generation. It is crucial to highlight that the speed bump identification is useful for several applications, such as routing problem when dealing with cost function parameterization, alert systems for car navigation task, traffic sign inventory, and to check if the speed bumps are in consonance to the local legislation.

These results are an essential step towards the improvement of ground transportation systems as a reliable way of alerting the drivers and the autonomous vehicles on the road via collaborative approach (V2V - vehicle-to-vehicle information network).

Future work considers the potential of other ML algorithms, as well as the improvement at the pre-processing stage and also the reading GoPro data as streaming and use a pre-trained static model generated by an ML algorithm to classify speed bumps in real-time.

Although our data acquisition base, mounted on top of a vehicle, is composed of multiples GoPro cameras, we are not using their images in this work. However, they will be used in the future to detect other features such as potholes and traffic signs (horizontal and vertical), allowing the map generation of all relevant components of a road, by using photogrammetric approach.

Acknowledgments

The authors would like to acknowledge the Universidade de Campinas for providing the used dataset, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP - grant number 2017/17003-3), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq under Grant 400699/2016-8), Universidade Federal de Uberlândia and University of New Brunswick for the research structure.

REFERENCES

- ALJAAFREH A, ALAWASA DK, ALJAAFREH S & ALABADLEH A. 2017. Fuzzy inference system for speed bumps detection using smart phone accelerometer sensor. *J Telecomun Electron Comput Eng* 9: 133-136.
- BELLO-SALAU H, AIBINU A, ONUMANYI A, ONWUKA E, DUKIYA J & OHIZE H. 2018. New road anomaly detection and characterization algorithm for autonomous vehicles. *Applied Comp and Info* 16(1/2): 223-239.
- CELAYA-PADILHA HM, GALVÁN-TEJADA CE, LÓPEZ-MONTEAGUDO FE, ALONSO-GONZÁLEZ O, MORENO-BÁEZ A, MARTINEZ-TORTEYA A, GALVÁN-TEJADA JI, ARCEO-OLAGUE JG, LUNA-GARCIA H & GAMBOA-ROSALES H. 2018. Speed bump detection using accelerometric features: A genetic algorithm approach. *Sensors* 18(2): 443 p.
- DANTAS GV. 2015. Utilização de classificador random forest na detecção de falhas em máquinas rotativas. (Unpublished data).
- DEVAPRIYA W, BABU CNK & SRIHARI T. 2015. Advance Driver Assistance System (ADAS) - Speed bump detection. *IEEE ICCIC*, 6 p.
- FERNÁNDEZ C, GAVILÁN M, LLORCA DF, PARRA I, QUINTERO R, LORENTE AG, VLACIC L & SOTELO MA. 2012. Free space and speed humps detection using lidar and vision for urban autonomous navigation. *IEEE Intelligent Vehicles Symposium*, p. 698-703.
- FFMPEG. 2018. URL: <https://www.ffmpeg.org/>. Accessed: 2018-09-20.
- GOPRO-UTILS. 2018. URL: <https://github.com/stilldavid/gopro-utils>. Accessed: 2018-09-20.
- HE G & ORVETS G. 2000. Capturing road network data using mobile mapping technology. *ISPRS Archives* 33(B2): 272-277.
- JULIA EDS. 2018. Um estudo comparativo entre algoritmos de classificação para fluxos contínuos de dados. (Unpublished data).
- MOLINA JSP, BLÁZQUEZ M & COLOMINA I. 2016. mapkite: A new paradigm for simultaneous aerial and terrestrial geodata acquisition and mapping. *ISPRS Archives* 41(B1): 957-962.
- SILVA IDS, SPATTI DH & FLAUZINO RA. 2010. *Redes Neurais Artificiais para engenharia e ciências aplicada*. Artliber Editora, 1ª edição.
- SOILÁN M, TRUONG-HONG L, RIVEIRO B & LAEFER D. 2018. Automatic extraction of road features in urban environments using dense als data. *Inter J Applied Earth Obs Geoinf* 64: 226-236.
- TAO CV & LIJ. 2007. *Advances in Mobile Mapping Technology*. Taylor and Francis. London: CRC Press, 9780415427234.
- VARMA VSKP, ADARSH S, RAMACHANDRAN KI & NAIR BB. 2018. Real Time Detection of Speed Hump/Bump and Distance Estimation with Deep Learning using GPU and ZED Stereo Camera. *Procedia Comp Science* 143: 988-997.
- WANG W, YANG N, ZHANG Y, WANG F, CAO T & EKLUND P. 2016. A review of road extraction from remote sensing images (English Edition). *J Traffic Transp Eng* 3: 271-282.
- WEKA. 2018. URL: <https://www.cs.waikato.ac.nz/ml/weka/>. Accessed: 2018-09-20.
- ZHAO J, YOU S & HUANG J. 2011. Rapid extraction and updating of road network from airborne lidar. *IEEE AIPR Workshop*, 7 p.

How to cite

MARQUES J, ALVES R, OLIVEIRA HC, MENDONÇA M & SOUZA JR. 2021. An evaluation of machine learning methods for speed-bump detection on a GoPro dataset. *An Acad Bras Cienc* 93: e20190734. DOI 10.1590/0001-3765202120190734.

*Manuscript received on July 1, 2019;
accepted for publication on November 18, 2019*

JOHNY MARQUES¹

<https://orcid.org/0000-0001-6148-158X>

RAULCEZAR ALVES¹

<https://orcid.org/0000-0001-5429-9595>

HENRIQUE C. OLIVEIRA²

<https://orcid.org/0000-0002-2783-4668>

MARCO MENDONÇA³

<https://orcid.org/0000-0003-0365-7895>

JEFFERSON R. SOUZA¹

<https://orcid.org/0000-0001-6422-4722>

¹Universidade Federal de Uberlândia, Faculdade de Computação, Av. João Naves de Ávila, 2121, Santa Mônica, 38400-902 Uberlandia, MG, Brazil

²Universidade de Campinas, Escola de Engenharia Civil, Arquitetura e Urbanismo, Rua Saturnino de Brito, 224, Cidade Universitária Zeferino Vaz, 13083-889 Campinas, SP, Brazil

³University of New Brunswick, Geodesy and Geomatics Engineering Department, 15 Dineen Dr, Fredericton, NB E3B 5H5, Canada

Correspondence to: **Johny Marques**

E-mail: johnymbr@gmail.com

Author contributions

The first author contributed with the original development of the methodology presented, experiments and writing the paper. The second author contributed to the experiments and writing the journal. The third and fourth authors contributed with mobile mapping system development, data collection and the writing the document. The fifth author contributed with initial writing and paper revisions.