Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

A Machine Learning Method for Vehicle Classification by Inductive Waveform Analysis ⋆

Bruno R. Vasconcellos * **Marcelo Rudek** ** **Marcelo de Souza** ***

* *Pontifícia Universidade Católica do Paraná - PUCPR, Control and Automation Egineering, Curitiba, PR 80215-901 (e-mail: brv1995@gmail.com).*
** *Pontifícia Universidade Católica do Paraná - PUCPR, Industrial and Systems Engineering Graduate Program, Curitiba, PR 80215-901 (e-mail: marcelo.rudek@pucpr.br).*
*** *Fiscaltech, Curitiba, PR 81290-270 (e-mail: marcelo.souza@fiscaltech.com.br)*

Abstract: The classification of vehicles is a matter of great importance for traffic control and management, helping with traffic surveillance as well as in statistical data collection. Among the several vehicular classification techniques, the most popular uses inductive loop sensors, because they achieve high accuracy rate at low cost. This paper proposes 5 different vehicle classification models by inductive waveform analysis: KNN, SVC, Decision Tree, Random Forest, and Voting Classifier. A brief introduction to the mathematical basis of these models and the main forms of vehicle detection are also presented. The obtained results reached an accuracy of 94% and showed how inductive waveform analysis is still a valid option for vehicle classification.

Keywords: Machine learning, Intelligent transportation systems, Information processing and decision support

1. INTRODUCTION

Vehicle detection and classification plays an important role in every intelligent transport management system. These processes help managing and supervising traffic and provide data for decision support.

According to the FHWA (2017), the vehicle detection systems are mainly devided into two categories: Pavement Invasive and Non-pavement Invasive Detectors. In developing countries, the most popular type of vehicle detector is the so-called Inductive Loop Detector (ILD). The main reason for that is its robustness at low cost (Almeida, 2010; Mohammed Ali et al., 2011). Although the ILDs are mainly used for speed calculation and traffic count, there are some systems that also perform the vehicle classification. In Brazil, companies that produce vehicle detection systems mostly use the method of calculating the vehicle's length to classify it. Normally these systems perform poorly. (Oliveira, 2011)

Image based classifications methods can reach to really good accuracy rate (Sundaravalli and Krishnaveni, 2018; Roecker et al., 2018), often they are not economically viable. In Brazil, the majority of fixed radars is based on inductive loops. To perform classification, it is more profitable to adapt these radars than to change the whole system. Vehicle classifiers based on ILD can reach an accuracy as good as the image based ones at a lower price. (Vasconcellos, 2019)

This paper shows 5 different machine learning methods for vehicle classification by ILDs waveform analysis.

2. BACKGROUD

The main goal of this paper is to select the best machine learning classification method based on IDLs. To do so, it is important to review the literature about this topic. There are several papers published approaching this problem, but most of them are based on artificial neural networks (ANN) (Hannan et al., 2015; Fazli et al., 2012; Harsha and Anne, 2016). Despite ANN having obtained good results, it is possible to achieve a high accuracy rate using a less powerful algorithm.

2.1 Inductive Loop Detector

An Inductive Loop Detector uses a coil installed under the pavement. When powered, the coil creates a magnetic field in the loop area. When a large metallic mass, like a vehicle, runs over the coil, a magnetic disturbance is caused.

If the initial and final period of this disturbance is timed, and the distance D between the loops is known, then the vehicle's speed can be measured by the equation 1. This configuration is known as speed trap. (Ki and Baik, 2006)

⋆ The authors would like to thank the Pontifical Catholic University of Parana for the technical support and the company Fiscaltech for providing all the data needed.

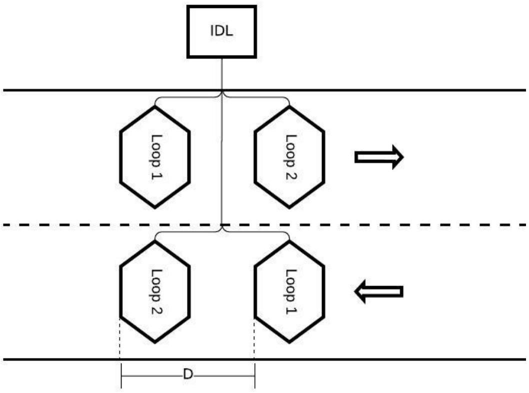

Fig. 1. A ILD using a speed trap configuration

$$V = \frac{1}{2}\left(\frac{D}{t_{2on} - t_{1on}} + \frac{D}{t_{2off} - t_{1off}}\right) \quad (1)$$

The vehicle's length can also be measured in a speed trap configuration. The equation 2 shows how to calculate the magnetic length of a vehicle.

$$L_{vehicle} = V_{vehicle}.t_{detection} - L_{loop}, \quad (2)$$

where $V_{vehicle}$ is the average speed of the vehicle, $t_{detection}$ is the time when the vehicle started running over the loop and L_{loop} is the length of the inductive loop. ILDs are often used to collect traffic volume, speed, vehicle classification, and weight information for pavement design, and much more. (Liao, 2018; Li et al., 2019)

The magnetic disturbance caused by the vehicle is often called a inductive loop signature technology (Jeng and Chu, 2014; Yogesh et al., 2018). This inductive waveform has a unique format for each type of vehicle, as shown in Fig 2.

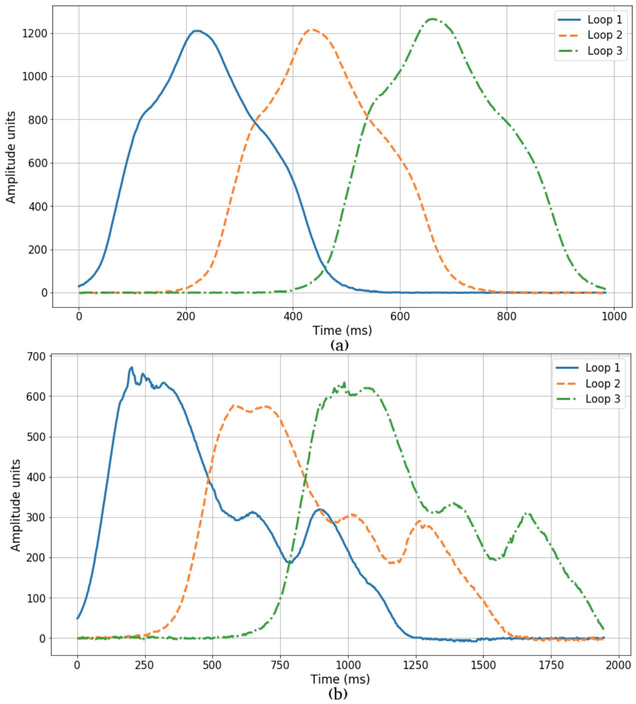

Fig. 2. Inductive waveform of a: (a) car, (b) truck

Acording to Ki and Baik (2006), there are several factors that define the inductive waveform format, like:

(1) The size, form, and conductivity of the vehicle;
(2) The 3D orientation of the vehicle when it runs over the loop;
(3) The size and form of the loop;
(4) The resonance frequency of the detection circuit;
(5) The distance between the vehicle and the loop.

2.2 K-Nearest Neighbors (KNN)

The KNN classifier is one of the simplest machine learning algorithms used as a classification method (James et al., 2013). It works by identifing the neighbors of a given observation. Given a positive integer K and a observation x_0, the KNN classifies by looking at the class of the K points in the training data that are closest to x_0, as shown in Fig. 3.

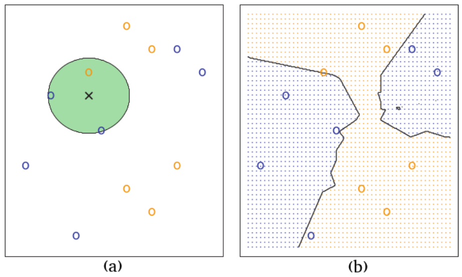

Fig. 3. (a) Test observation and their neighbors. (b) Decision region for K = 3

The prediction is made by counting the most frequent class of the test observation's neighbors, in the case of Fig. 3, the prediction would be Blue. All the data must be in the same scale so the KNN classifier can be applied.

2.3 Support Vector Classifier (SVC)

The SVC classifier is based on Support Vector Machiens (SVM), where a dimensional space of features is created. These features are then separated into classes by a hyper-plane in space. The observations that define the limits of the hyperplane are called *support vectors*, and they are the ones that "support" the position of the hyperplane in space (James et al., 2013). The SVMs can have a linear kernel when the classes are linearly separable and a non-linear kernel when they are not.

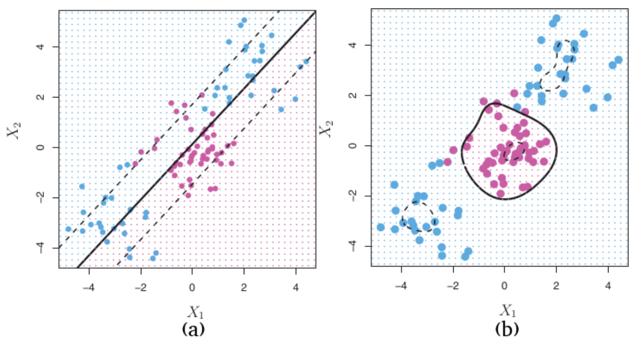

Fig. 4. (a) Linear kernel (b) Radial kernel

Fig. 4 (a) shows an example of a linear kernel SVM attempting to divide two classes that are not linearly separables, and Fig. 4 (b) shows the same observations with a radial kernel SVM.

An important tunning parameter that must be defined when using the SVC is the value of C. C is a non-negative variable that allows individual observations to be on the wrong side of the hyperplane. The larger the C, the more tolerant the SVC becomes in therms of observation violation (Hastie et al., 2009). Just as it is the case for the KNN classifier, all data must be in scale.

2.4 Decision Trees

Decision Trees are classification and regression methods that present a structure similar to a flowchart. They are formed by a *root node* that is divided into *nodes* by *branches*. All the possible predictions are called *leaves*. They usually are not as robust as the other supervised methods mentioned earlier, but they are really easy to implement, mainly because the data doesn't need to be preprocessed and they are easy to visualize.

A really important parameter when designing a decision tree is the impurity criterion. This parameter works as a cost function when spliting the nodes. One commonly used is the *Gini index*, given by the equation 3.

$$G = \sum_{k=1}^{K} P_k(1 - P_k), \tag{3}$$

where K is number of classes and P_k is the probability of randomly choosing an element of class k. The higher the *Gini index*, the better the split. (Hastie et al., 2009)

3. PROPOSED METHOD

The proposed method consists of executing the following tasks: acquiring a vehicle's inductive waveform dataset, labeling the dataset with the vehicle classes, analyzing the data to choose which features to extract, extracting the features, testing the classifiers and comparing the results, as Fig. 5 shows.

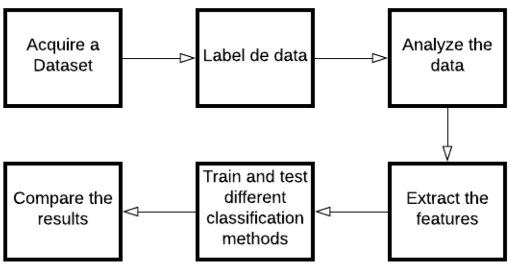

Fig. 5. Flow-chart of the process

The vehicles categories were divided into 5 classes, as Table 1 shows.

A dataset of 571 labeled inductive waveforms was used in this paper, where 107 were from motorcycles, 115 from cars, 134 from pickups and vans, 109 from trucks and 106 from buses.

Table 1. Vehicle classes

Class	Example
Motorcycle	motorcycles and scooters
Car	passenger cars
Pickup	pickup trucks and vans
Truck	trucks for urban applications
Bus	buses

After analyzing the data and based on the literature review, as Calixto (2006), Almeida (2010) and Oliveira (2011), 10 features were chosen that best represent each class inductive waveform:

1. Max Amplitude Value: is the max amplitude value of the waveform and is higher for vehicles closer to the ground;
2. Inductive Waveform Mean: is the mean of the waveform and is higher in vehicles with a high amplitude;
3. Inductive Waveform Variance: the waveform variance helps to differentiate vehicles with a similar waveform mean;
4. Inductive Waveform Standard Deviation: same as the variance, helps to differentiate the classes even more;
5. Kurtosis: gives information about the shape of the distribution;
6. Skewness: gives information about the assimetry of the distribution;
7. Number of Peaks: the number of peaks of the waveform;
8. Mean of Peak Value: the mean value of all peaks helps distinguishing vehicles with a same number of peaks.
9. Inductive Waveform Area: is the area under the waveform curve;
10. Magnetic Length: the length of the vehicle calculated with the waveform.

All the features were extracted from the first inductive loop. Fig. 6 shows an example of a violin distribution of the number of peaks of the inductive waveform for each class. Because larger vehicles tend to stay longer over each loop, and due to the fact that their metalic mass is not uniform throughout the whole body, the inductive waveform presents various peaks.

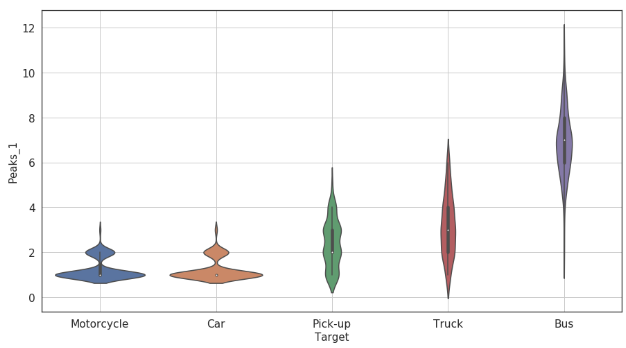

Fig. 6. Number of peaks for each class

On the other hand, smaller vehicles like motorcycles and cars present only 1 or 2 peaks.

3.1 Training and Testing

All methods were trained splitting the dataset into 70% *train data* and 30% *test data* and all data was stratified to ensure a balance between the classes.

The first machine learning algorithm tested was the KNN classifier. Due to the fact that the KNN is a classifier based on the Euclidean Distance between the points, all the data must be in the same scale. For that matter, all the data was standarized using the scikitlearn function *StandardScaler*. (Pedregosa et al., 2011).

The best value of K that was found was 14, giving the following results:

Table 2. KNN score

Class	precision	recall	f1-score
Motorcycle	97%	97%	97%
Car	79%	94%	86%
Pickup	86%	75%	80%
Truck	88%	88%	88%
Bus	100%	94%	97%
Accuracy			89%

The KNN method presented a good precision rate for buses, reaching 100%, while for cars it only reached 79%.

The second method tested was the SVC. Just as the KNN, the SVC was also standarized to achieve a better result. The chosen parameters were $C = 100$ and *radial* kernel. The SVC reached a more balanced result for all the classes then the KNN method.

Table 3. SVC score

Class	precision	recall	f1-score
Motorcycle	97%	100%	98%
Car	87%	97%	92%
Pickup	86%	80%	83%
Truck	86%	76%	81%
Bus	91%	97%	94%
Accuracy			90%

The Decision Tree was the third method tested. The *Gini index* was used as criterion, resulting the following scores:

Table 4. Decision Tree score

Class	precision	recall	f1-score
Motorcycle	100%	100%	100%
Car	72%	83%	77%
Pickup	76%	70%	73%
Truck	93%	76%	83%
Bus	86%	97%	91%
Accuracy			84%

The next classifier tested was a Random Forest classifier. The Random Forest consited of 200 decision trees using the *Gini index* criterion. Fig. 5 shows the results.

Table 5. Random Forest score

Class	precision	recall	f1-score
Motorcycle	100%	100%	100%
Car	78%	91%	84%
Pickup	89%	78%	83%
Truck	97%	85%	90%
Bus	89%	97%	93%
Accuracy			90%

The fifth and last classifier tested was a Voting Classifier composed of a KNN, SVC and Random Forest. The Voting Classifier combines different methods and uses the average predicted probabilities to predict the Target. When weights are provided, the Voting Classifier multiplies each classifier probability with their respective weight. The weights given for each classifier were 4 for the Random Forest, 2 for the SVC and 1 for the KNN. The results can be seen in Table 6.

Table 6. Voting Classifier score

Class	precision	recall	f1-score
Motorcycle	100%	100%	100%
Car	94%	91%	93%
Pickup	86%	93%	89%
Truck	94%	88%	91%
Bus	97%	97%	97%
Accuracy			94%

4. RESULTS AND DISCUSSION

When analyzing each classifier score, it becomes clear that some methods had achieved better results when predicting a certain class than others, e.g. the KNN classifier had a lower precision when predicting cars than the SVC. On the other hand, it had a 100% precision rate classifing buses, while the Decision Tree only had 86%. Uniting these advantages that one classifier has over the other is what makes the Voting Classifier so powerful.

To measure the feature extraction and prediction latency, the Voting Classifier was tested in 571 vehicle inductive waveforms. It took 15.417 seconds to extract the featuers and predict all the data, an average of 0.027 seconds per vehicle with a 94% of accuracy rate.

One advantage that the Random Forest has is the ability to evaluate the feature importances, as Fig 7 shows.

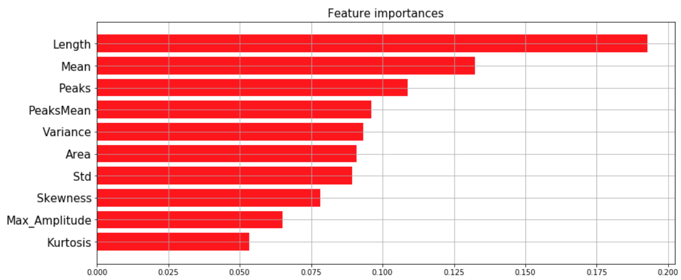

Fig. 7. Feature Importance

To reduce computacional costs, a new Voting Classifier was trained using only the better half of the features given in Fig 7. For the same 571 vehicles dataset, it took 8.907 seconds to extract the features and predict, and average of 0.0156 seconds per vehicle, 25% faster than with 10 features but with a 2% drop in accuracy.

Table 7. 5 and 10 features comparative

Feat.	Motorcycle	Car	Pickup	Truck	Bus	Latency
10	100%	94%	86%	94%	97%	20.7ms
5	97%	91%	87%	86%	100%	15.6ms

To improve the accuracy even more, another method was tested analyzing each individual loop. The data used in this paper contained information about all three loops of the ILD. Therefore it was possible to multiply the class probability for each inductive loop output. Table 8 shows an example of a observation where the classifier predicted a Truck for the first and second loop, but a Pickup for the third loop.

Table 8. Predictions for each loop

Pred.	Motorcycle	Car	Pickup	Truck	Bus
1	0.0027	0.0083	0.4102	0.5781	0.0006
2	0.0023	0.0052	0.1826	0.8087	0.0010
3	0.0022	0.0064	0.5310	0.4590	0.0010
Final	$1,36.10^{-7}$	$2,76.10^{-6}$	$3,98.10^{-1}$	2.15	6.10^{-9}

Using this method, the final prediction would be the class with the highest value, in this case, a Truck. Although this method can increase the accuracy a bit, it makes the system three times slower, because it needs to extract the features for all three loops. For the data tested, only 9 out of 571 predictions weren't the same for all loops.

5. CONCLUSION

This paper shown 5 different machine learning algorithms for vehicle classification. The best method tested was a Voting Classifier composed of a KNN, SVC and, Random Forest. The overall accuracy achieved 94%. The results showed how ILDs can still be used as an efficient automatic system for traffic surveillance and anonymous data collection

The proposed system can be implemented in places where ILDs are still the primary method of vehicle detection, without the need to change the current hardware. The system can be used for statistical data collection, assisting in roads maintenance planning and expansion. It can also be used for surveillance, e.g. for monitoring highways where there are different speed limits for different vehicle types or monitoring lanes where only a exclusive vehicle type is allowed, among other applications.

REFERENCES

Almeida, F.A.M. (2010). *Classificação automática de veículos pelo perfil Magnético através de técnicas de aprendizagem de máquina*. Master's thesis, Universidade Estadual do Ceará.

Calixto, S.A. (2006). *Classificação de veículos através de sistemas fuzzy*. Master's thesis, Universidade Tecnológica Federal do Paraná.

Fazli, S., Mohammadi, S., and Rahmani, M. (2012). Neural network based vehicle classification for intelligent traffic control. *International Journal of Software Engineering & Applications*, 3.

FHWA (2017). Traffic control system handbook.

Hannan, M.A., GEE, C., and Javadi, S. (2015). Automatic vehicle classification using fast neural network and classical neural network for traffic monitoring. *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, 23, 2031–2042. doi:10.3906/elk-1211-46.

Harsha, S.S. and Anne, K.R. (2016). Gaussian mixture model and deep neural network based vehicle detection and classification. *International Journal of Advanced Computer Science and Applications*, 7(9). doi:10.14569/IJACSA.2016.070903. URL `http://dx.doi.org/10.14569/IJACSA.2016.070903`.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer.

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*. Springer.

Jeng, S.T. and Chu, L. (2014). A high-definition traffic performance monitoring system with the inductive loop detector signature technology. *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 1820–1825.

Ki, Y.K. and Baik, D.K. (2006). Vehicle-classification algorithm for single-loop detectors using neural networks. *IEEE Transactions on Vehicular Technology*, 55, 1704–1711.

Li, Y., Tok, A.Y.C., and Ritchie, S.G. (2019). Individual truck speed estimation from advanced single inductive loops. *Transportation Research Record*, 2673(5), 272–284. doi:10.1177/0361198119841289.

Liao, C.F. (2018). Investigating inductive loop signature technology for statewide vehicle classification counts.

Mohammed Ali, S., George, B., Vanajakshi, L., Jayashankar, V., and Kumar, J. (2011). A multiple inductive loop vehicle detection system for heterogeneous and lane-less traffic. *Instrumentation and Measurement, IEEE Transactions on*, 61, 1–5. doi:10.1109/IMTC.2011.5944278.

Oliveira, H.A. (2011). *Sistema de reconhecimento de padrões para identificação de porte de veículos através de análise de perfil magnético*. Master's thesis, Universidade Federal do Ceará.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

Roecker, M., Costa, Y., Almeida, J., and Matsushita, G. (2018). Automatic vehicle type classification with convolutional neural networks. 1–5. doi: 10.1109/IWSSIP.2018.8439406.

Sundaravalli, G. and Krishnaveni, K.S. (2018). March 2018 a survey on vehicle classification techniques.

Vasconcellos, B.R. (2019). *Propósta de um método de aprendizagem de máquina para classificação de veículos por análise de perfil magnético*. Bachelor's thesis, Pontifícia Universidade Católica do Paraná.

Yogesh, G.K.V., Sharma, A., and Vanajakshi, L. (2018). An improved inductive loop detector design for efficient traffic signal operations and leaner space requirements. *Transportation Research Record*, 2672(18), 143–153. doi: 10.1177/0361198118798457.