Proceedings of 4th International Conference on Information Society and Technology (ICIST 2014), Kopaonik, Serbia, March 9-13
ISBN:978-86-85525-14-8

Vehicle Classification and False Detection Filtering using a Single Magnetic Detector based Intelligent Sensor

Peter Sarcevic*, Szilveszter Pletl**
* University of Szeged, Department of Informatics, Szeged, Hungary
** University of Szeged, Department of Informatics, Szeged, Hungary
** University of Novi Sad, Novi Sad, Serbia
sarcevic@inf.u-szeged.hu, pletl@inf.u-szeged.hu

***Abstract*— Vehicle detection and classification is a very actual problem, because vehicle count and classification data are important inputs for traffic operation, pavement design, transportation planning and other applications. Magnetic detector based sensors provide many advantages compared to other technologies. In this work a new vehicle detection and classification method is presented using a single magnetic detector based system. Due to the relatively big number of false detections caused by vehicles with high metallic content passing in the neighboring lane, a technique for false detection filtering is also presented. Vehicle classes are determined using a feedforward neural network which is implemented in the microcontroller of the detector, together with the detection algorithm and the algorithm used for determining the neural network inputs. The gathering of training samples and testing of the trained neural network have been done in real environment. For the training of the neural network the back-propagation algorithm has been used with different training parameters.**

Keywords*— vehicle detection, false detection filtering, vehicle classification, magnetic sensors, neural networks

I. Introduction

To provide speed monitoring, traffic counting, presence detection, headway measurement, vehicle classification, and weigh-in-motion data, the need for automatic traffic monitoring is increasing. This urges the manufacturers and researchers to develop new technologies and improve the existing ones. Vehicle count and classification data are important inputs for traffic operation, pavement design, and transportation planning. In traffic control, signal priority can be given to vehicles classified as bus or an emergency vehicle.

In this work, a new detection and classification method for a single magnetic sensor based system is discussed, and a technique for filtering the false detections caused by vehicles passing in the neighboring lane is also presented.

Magnetic sensors can measure the changes in the Earth's magnetic field caused by the presence of metallic objects. Magnetic vehicle detectors use the changes generated by the metallic content of vehicles when they are near the sensor as written in Reference [1]. Two sensor nodes placed a few feet apart can estimate speed as described in Reference [2]. A vehicle's magnetic 'signature' can be processed for classification.

Advantages and disadvantages of magnetic detectors are shown in Table 1.

This work was supported by TÁMOP-4.2.2.A-11/1/KONV-2012-0073

II. The single magnetic detector based intelligent sensor

The used magnetic sensor is an HMC5843 based unit developed by "SELMA" Ltd. and "SELMA Electronic Corp" Ltd., companies from Subotica, Serbia. Two types of magnetic detectors have been developed, one with cable and one with wireless communication.

For classification sample collection and for detection and classification efficiency testing, a unit with cable communication has been mounted in Subotica, on the main road passing through the town. All vehicles classes can be found passing on the mentioned road, so the place is ideal. The sensor has been mounted 5 centimeters beneath the pavement surface. The direction of the axis is very important, because the network inputs are calculated by axis, and if the positioning is changed, the waveforms will not be the same. The X axis points to the movement direction, the Y axis points to the neighboring lane, and Z is orthogonal with the pavement surface.

The Honeywell HMC5843 is a small (4x4x1.3 mm) surface mount multi-chip module designed for low field magnetic sensing. The 3-axis magnetoresistive sensors feature precision in-axis sensitivity and linearity, solid-state construction with very low cross-axis sensitivity designed to measure both direction and magnitude of Earth's magnetic fields, from tens of micro-gauss to 6 gauss. The highest sampling frequency is 50Hz.

Wireless magnetic sensor networks offer an attractive,

TABLE I.
ADVANTAGES AND DISADVANTAGES OF MAGNETIC SENSOR BASED VEHICLE DETECTORS

Advantages	Disadvantages
• Insensitive to inclement weather such as snow, rain, and fog • Less susceptible than loops to stresses of traffic • Some models transmit data over wireless RF link • Some models can be installed above roads, no need for pavement cuts	• Difficult to detect stopped vehicles • Installation requires pavement cut or tunneling under roadway • Decreases pavement life • Installation and maintenance require lane closure

low-cost alternative to inductive loops, video and radar for traffic surveillance on freeways, at intersections and in parking lots as written in Reference [3].

III. VEHICLE DETECTION ALGORITHM

As written in References [4] and [5], magnetic detectors are capable of very high, above 97 percent detection accuracy with proper algorithms. In Reference [6] 97% of detection accuracy has been achieved using neural networks and fuzzy data fusion. Most of the algorithms use adaptive thresholds as used in Reference [7].

In Reference [4] the effect of temperature on HMC magnetic sensor measurements is described. The temperature on the pavement can change a lot in the course of a day, but the changes in the measured values are very slow.

The developed vehicle detection algorithm uses thresholds which can change when no detection is available to avoid the effects of temperature changes. The principles of the algorithm:

- A calibration process is run when the unit is turned on. Maximum and minimum values are determined in a period of time at all three axis (if even at one axis the difference between the maximum and minimum exceeds a previously defined width, the calibration starts from the beginning). After this stage, the range is equally stretched to the previously defined width, and the upper and lower thresholds are now determined at all three axis. This method makes the further algorithm immune to noise.
- If the measures exceed the range determined by the thresholds at axis X and Z, detection is generated (detection flag is “1”). If only one axis exceeds the range, probably a vehicle is passing in the neighboring lane.
- In case of detection, the detection flag goes back to “0” if measures in all three axis are between thresholds for a previously defined number of measures.
- If measures at all three axis are in the range determined by the thresholds, and no detection is available, the algorithm calculates new thresholds.

The axis along the direction of travel can be used to determine the direction of the vehicle, what is shown in Reference [8]. When there is no car present, the sensor will output the background earth’s magnetic field as its initial value. As the car approaches, the earth’s magnetic field lines of flux will be drawn toward the ferrous vehicle.

A. Efficiency

For testing the efficiency of the algorithm a one hour test has been done. The results have been divided by vehicle classes, and are shown in Table 2. As seen, the algorithm is effective, only motorcycles can cause failures. The reason of failures in motorcycle detection can be the low metallic content, and the distance from the detector.

As the results show, the number of false detections is high. This is caused by vehicles passing in the neighboring lane with high metallic content, usually trucks or buses. Filtering a part of these detections could be done by increasing the width of the detection ranges, but this can affect the motorcycle detection efficiency, and the classification algorithm could lose important parts of the waveforms.

TABLE II.
EFFICIENCY OF THE DETECTION ALGORITHM DURING A ONE HOUR TEST

Vehicle class	Passed	Detected	Rate
Motorcycle	4	3	75%
Car	168	168	100%
Van	10	10	100%
Truck	15	15	100%
Bus	6	6	100%
Other	2	2	100%
False detections caused by vehicles passing in the neighboring lane		13	
Σ	205	217	94,15%

IV. SAMPLE COLLECTION

For neural network training and false detection filtering samples have been collected using the mounted sensor. The measurement values and a detection number, which has been incremented at every rising edge of the detection flag, have been saved into a database.

To declare the classes (neural network targets) of the passing vehicles, and to separate the good and false detections, we used the images made by a camera mounted beside the road. The images have been saved at every falling edge of the detection flag, and have been named using the detection number.

Altogether measures of 11021 passing vehicles had been collected.

V. FALSE DETECTION FILTERING

The gathered samples had been divided into 3 groups using the images: good detections (10218 samples), false detections (345 samples), and vehicles passing between the two lanes (458 samples), which are also false detections.

The basic idea of the filtering algorithm was to generate different rules, and to find optimal parameter values with which false detections can be determined. 18 rule types have been declared, and the optimization has been done for all types using specific parameters calculated of every sample.

The optimizations have been done using genetic algorithms. Every optimization has been done in 3000 generations with a population size of 50. The fitness functions determined the rate of misses at all samples. If the result of the rule is “true”, the detection is false.

The used rules are shown in Table 3, where X, Y and Z are measurement values in the samples, Ylth is the number of measures where Y was continuously not between thresholds before the detection was declared, and YX, YZ, XY, Xd, Yd, Zd and YL are the optimized parameters. In types where X/Z has been used, the fitness function chooses between “<” and “>” using a further parameter. The rules which consist j, have j number of same parts with “or” between them, so if one part gives as result “true”, the output of the whole rule is “true”. This way the functions were able to filter out more groups of samples which have similar values. Optimization has been done at every type with j={1,2,…,10}.

TABLE III.
ADVANTAGES AND DISADVANTAGES OF MAGNETIC SENSOR BASED VEHICLE DETECTORS

Number	Rule
1	$((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (X/Z <> XZ(j)))_j$
2	$((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (X/Z <> XZ(j) \text{ and } (Ylth > YL(j)))_j$
3	$((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)))_j$
4	$((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (Ylth > YL(j)))_j$
5	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Ylth > YL_b)$
6	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Ylth > YL_b) \text{ or } ((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (X/Z <> XZ(j)))_j$
7	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Ylth > YL_b) \text{ or } ((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (X/Z <> XZ(j) \text{ and } (Ylth > YL(j)))_j$
8	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Ylth > YL_b) \text{ or } ((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)))_j$
9	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Ylth > YL_b) \text{ or } ((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (Ylth > YL(j)))_j$
10	$((X < Xd(j)) \text{ and } (Y > Yd(j)) \text{ and } (Z < Zd(j)))_j$
11	$((X < Xd(j)) \text{ and } (Y > Yd(j)) \text{ and } (Z < Zd(j) \text{ and } (Ylth > YL(j)))_j$
12	$(Y > Yd_b) \text{ or } (Ylth > YL_b)$
13	$(Y > Yd_b) \text{ or } (Ylth > YL_b) \text{ or } ((X < Xd(j)) \text{ and } (Y > Yd(j)) \text{ and } (Z < Zd(j)))_j$
14	$(Y > Yd_b) \text{ or } (Ylth > YL_b) \text{ or } ((X < Xd(j)) \text{ and } (Y > Yd(j)) \text{ and } (Z < Zd(j) \text{ and } (Ylth > YL(j)))_j$
15	$((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (Y > Yd(j)))_j$
16	$((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (Y > Yd(j)) \text{ and } (Ylth > YL(j)))_j$
17	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Y > Yd_b) \text{ or } (Ylth > YL_b) \text{ or } ((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (Y > Yd(j)))_j$
18	$(Y/X > YX_b) \text{ or } (Y/Z > YZ_b) \text{ or } (Y > Yd_b) \text{ or } (Ylth > YL_b) \text{ or } ((Y/X > YX(j)) \text{ and } (Y/Z > YZ(j)) \text{ and } (Y > Yd(j)) \text{ and } (Ylth > YL(j)))_j$

X, Y and Z values are the distances between the measurement values at a specific point and the ranges specified by the thresholds.

To try to declare false detections immediately, at the start of the detection, the optimizations have been done with X, Y and Z values calculated of the measurement values in the moment when X and Z first exceeded their ranges. The results showed that with this method almost none of the false detections can be filtered without declaring good detections as false. This is because the X and Z axis exceed their ranges too quickly to see bigger differences in Y. Range widths could be increased, but this would result information loss at the classification algorithm.

As the false detections can not be declared immediately, the optimizations have been done with the highest X, Y and Z values during the entire detection. The results are shown on Fig.1. It can be seen, that the results are very similar, but the optimizations could not reach any usable parameters at types 13 and 14, where the optimized values were very small, and gave bad results for all good detections.

The best results at all cases had been reached with type 10, which achieved recognition rates of 95.9% all detections, 98.85% at good, 75.94% at false and 44.98% at detections where the vehicle passed between the two lanes, which were also declared as false. This means that 58.28% of false detections have been filtered.

The loss of around 1% of good detections could be the result of cases when a vehicle was passing in the neigboring lane with high metallic content beside the vehicle which should be detected.

The results showed that types which were tested with different *j* values did not differ greatly with the added futher parts, only small improvements could be noticed when *j* was bigger then 1.

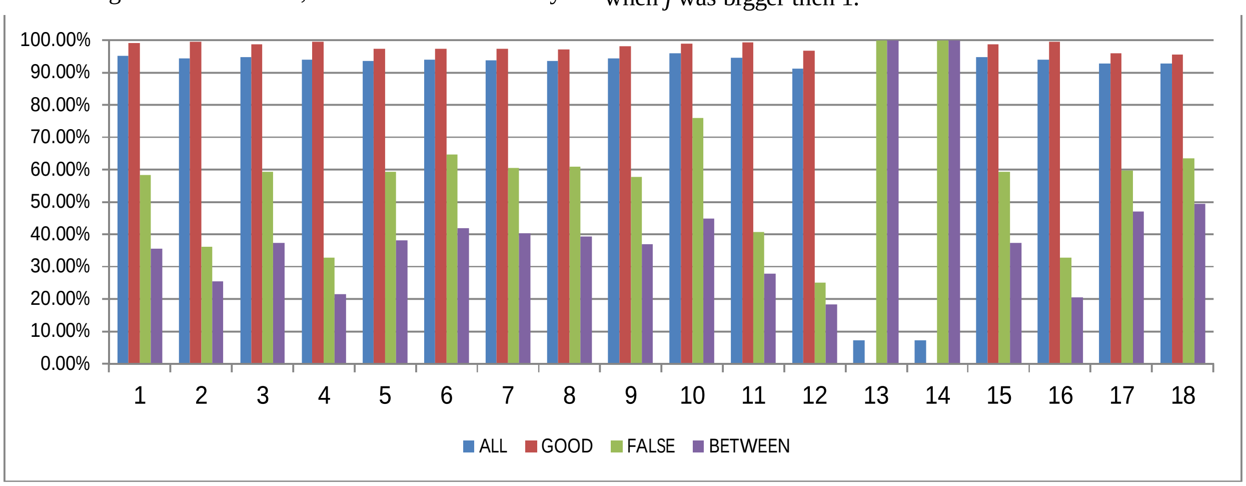

Figure 1. Hit rates of false detection filtering after optimization of parameters for different rule types

VI. Classification algorithm

The basic idea was to collect the measurement values when the detection flag is "1", and calculate specific parameters from the magnetic signature, which can be applied to the inputs of the neural network.

A. Other classification algorithms

Classification stations with highly calibrated inductive loops are very popular. However, the infrastructure and maintenance costs of such a vehicle classification station are high.

In Reference [9] an artificial neural network based method was developed to estimate classified vehicle volumes directly from single-loop measurements. They used a simple three-layer neural network with back-propagation structure, which produced reliable estimates of classified vehicle volumes under various traffic conditions. In this study four classes (by ranges of length) were defined, and all classes had an own ANN. All networks had 19 nodes in the input layer, 1 node for the time stamp input and 9 pairs of nodes for inputting single-loop measurements (volume and lane occupancy). All networks had one output node (each was one class bin), but the number of hidden neurons differed for each class (35 for class1, 8 for class2, 5 for class3 and 21 for class4).

Sun, in Reference [10] studied the use of existing infrastructure of loop detectors for vehicle classification with two distinct methods. The seven-class scheme was used for the first method because it targets vehicle classes that are not differentiable with current techniques based on axle counting. Its first method uses a heuristic discriminant algorithm for classification and multi-objective optimization for training the heuristic algorithm. Feature vectors obtained by processing inductive signatures are used as inputs into the classification algorithm. Three different heuristic algorithms were developed and an overall classification rate of 90% was achieved. Its second method uses Self-Organizing Feature Maps (SOFM) with the inductive signature as input. An overall classification rate of 80% was achieved with the four-class scheme.

In the last few years a big number of studies have been made with classification algorithms using magnetic sensors.

The rate of change of consecutive samples was compared with a threshold in Reference [11] and declared to be +1 (–1) if it is positive and larger than (negative with magnitude larger than) the threshold, or 0 if the magnitude of the rate is smaller than the threshold. The second piece of information was the magnetic length of the vehicle. 82% efficiency was achieved, with vehicles classified into five classes.

Reference [12] achieved a vehicle detection rate better than 99 percent (100 percent for vehicles other than motorcycles), estimates of average vehicle length and speed better than 90 percent, and correct classification into six types with around 60 percent, when length was not used as a feature.

In Reference [13], with x and z dimension data and without vehicle length information, a single magnetic sensor system, with a Multi-Layer Perceptron Neural Network, 93.5 percent classification efficiency was achieved, but vehicles were only separated into two classes. In a double sensor system 10 classes were selected for development, and 73.6 percent was achieved with length estimation and a methodology using K-means Clustering and Discriminant Analysis.

B. The used neural network and the input parameters

A three-layer feedforward neural network has been used for vehicle class estimation. The neurons in the hidden layer have logarithmic sigmoid transfer functions, while the output layer neurons use saturating linear functions. The structure of the used neural network can be seen on Fig.2. Bias values have not been used in the network because the network has to be implementable in a neural network, and the bias values would need big memory space.

The networks have been trained using the backpropagation algorithm. During the training, weights have been modified after every sample. Using this network the error of the output layer output can be calculated with the next formula (1):

$$\delta_\alpha = \mathrm{Target}_\alpha - \mathrm{out}_\alpha \tag{1}$$

where δ_α is the error of α output neuron, $Target_\alpha$ is the target value, and out_α is the current output of the neuron.

The output neuron weights have to be modified the next way (2):

$$W_{A\alpha}^{+} = W_{A\alpha} + \eta * \delta_\alpha * \mathrm{out}_A \tag{2}$$

where $W_{A\alpha}^{+}$ is the modified weight between A hidden and α output neuron, and η is the learning rate.

The error of the hidden layer neurons can be calculated using the errors of the output neurons, the weights between the hidden neuron and each output neuron, and the output of the hidden neuron (3):

$$\delta_A = \mathrm{out}_A * (1 - \mathrm{out}_A) * (\delta_\alpha * W_{A\alpha} + \delta_\beta * W_{A\beta}) \tag{3}$$

The modifications of the weights between the input and the hidden layer can be done with the next formula (4):

$$W_{\lambda A}^{+} = W_{\lambda A} + \eta * \delta_A * \mathrm{in}_\lambda \tag{4}$$

where in_λ is the input of the λ input neuron.

Network training has been done with different number of neurons in the hidden layer, and different learning rates.

The network has 6 outputs, because 6 vehicle classes had been defined to be classified: motorcycles, cars, vans, trucks, buses and other. The class with the biggest output will be declared as the class of the passed vehicle.

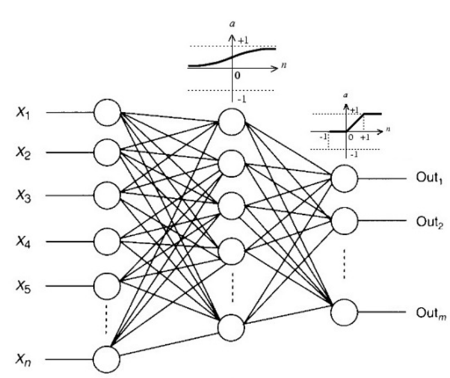

Figure 2. The structure of the used neural network

The input layer consists 16 neurons. These are parameters calculated of the waveforms at each axis. The network inputs are the next:

- 1 input – Detection length (number of measures made while the detection flag is "1")
- 6 inputs - The biggest differences between measured values and thresholds at each axis (the difference between the highest measured value and the upper threshold (5), and the difference between the lower threshold and the smallest measured value (6))

$$X_{upper\_diff} = X_{highest\_val} - X_{upper\_thr} \quad (5)$$

$$X_{lower\_diff} = X_{lower\_thr} - X_{lowest\_val} \quad (6)$$

- 6 inputs – Number of local maximums (if the measured values are above the upper threshold), and local minimums (if the values are under the lower threshold) at each axis.
- Range changes at each axis. The thresholds define three ranges, one above the upper threshold, one under the lower, and one between them.

C. Neural network training

Measurement data for 130 samples per class have been collected for network training.

The network training has been done in three series depending on the learning rates of the layers. The used rates were the next:

- 0.11 at the output and 0.1 at the hidden layer
- 0.08 at the output and 0.06 at the hidden layer
- 0.05 at the output and 0.04 at the hidden layer

Every learning rate pair has been tested with 1 to 25 hidden layer neurons. Every of the 75 trainings has been done in 1000 epochs.

Of 130 samples per class, 90 have been used for training, and 40 for validation.

During the training process the matching rates and the mean squared errors at training and validation samples have been calculated after every sample in the epoch. The highest matching rates and the smallest mean squared errors have been saved. When finding a better value of each parameter, all current values of other performance parameters have also been saved together with a matrix containing the current places of the misses. The current number of the iteration and the sample number have also been recorded to see are more iterations needed.

The highest matching rates with all learning rate pares depending on the number of hidden layer neurons can be viewed on Fig.3. The efficiency of the network both on training and validation samples has improved when learning rates have been reduced. The highest matching rate on training samples, 88.44%, has been recorded with 18 hidden layer neurons. The highest efficiency on validation samples was 70.83%. The rate at training samples can be improved by increasing the number of iterations, because at most cases the biggest value was achieved near to the end of the training process. But this is not true for the validation samples, where in most of the cases the maximums were recorded around 300 epochs.

On Fig.4 the smallest mean squared errors are shown found during the trainings. As seen, the smallest values were achieved also with the smallest learning rate pair. The values similarly as at matching rates, could have been improved at training samples by using a longer training process, but the smallest values at validation samples were also recorded around 300 iterations.

The number of misses by classes in the case when the highest matching rate at training samples has been found are shown in Table 5 for training samples and Table 4 for validation samples. The places with most misses are very similar. The most misses are were made between classes 2, 3 and 4. A possible reason can be that cars, vans and smaller trucks are almost the same length, the number of axles is also the same, and the distance between them is also similar.

D. Neural network implementation and testing

During the implementation a very important factor was not to stop the measuring for the time of the network output calculating. The network input calculation and updating has been done after every measurement when the detection flag was "1".

After a falling edge on the detection flag, the network outputs are calculated, and the vehicle class is determined. This process is done until the next measurement is made, so the class is determined in 20ms, what is a measurement cycle.

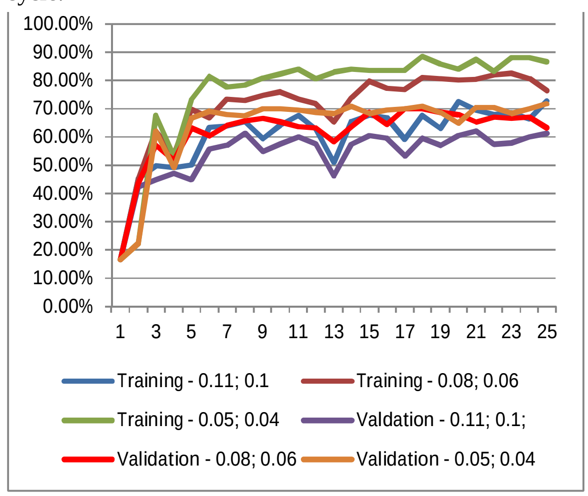

Figure 3. Highest matching rates achieved at training and validation samples with different number of hidden layer neurons and different learning rates.

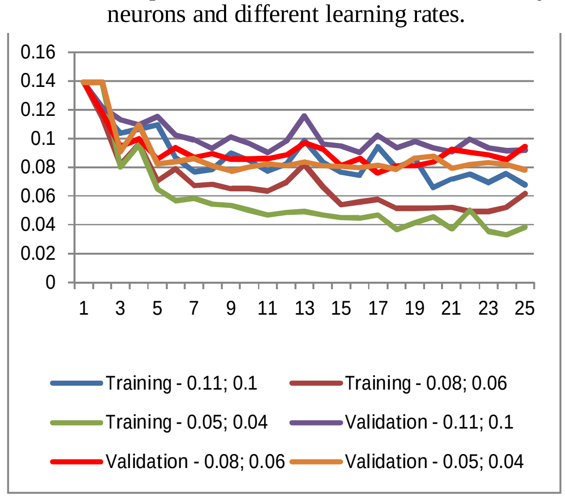

Figure 4. Smallest mean squared errors achieved at training and validation samples with different number of hidden layer neurons and different learning rates.

TABLE IV.
PLACES AND NUMBER OF MISSES AT VALIDATION SAMPLES

Output \ Target	1.	2.	3.	4.	5.	6.	Rate
1.	0	1	1	0	0	3	87,5%
2.	1	0	9	9	0	0	52,5%
3.	0	11	0	10	1	1	42,5%
4.	3	7	5	0	9	1	37,5%
5.	0	4	2	6	0	0	70%
6.	2	0	0	0	0	0	95%

TABLE V.
PLACES AND NUMBER OF MISSES AT TRAINING SAMPLES

Output \ Target	1.	2.	3.	4.	5.	6.	Rate
1.	0	1	1	0	0	0	97,33%
2.	1	0	9	7	0	0	77,33%
3.	1	7	0	6	1	0	80%
4.	2	4	4	0	4	1	80%
5.	0	2	1	0	0	0	96%
6.	0	0	0	0	0	0	100%

For testing, the weights of the 18 hidden neuron network has been implemented which achieved the highest matching rate on training samples. This network had 64.17% efficiency on validation data, the mean squared error of the training samples was 0.041736, and 0.096034 of the validation samples.

The network was tested for 300 detections, and the results are shown in Table 6. As seen, the recognition rates are very similar to the values calculated on validation samples during training (Table 4). The testing was not ideal, because some classes had very small number of vehicles passing in this testing interval. During this test period the false detection filtering has not been used.

TABLE VI.
TEST RESULTS OF THE IMPLEMENTED NEURAL NETWORK

	1.	2.	3.	4.	5.	6.	Σ	Recognition rate
1.	2	0	0	0	0	0	2	100%
2.	18	99	48	55	5	4	229	43,23%
3.	0	1	6	5	0	0	12	50%
4.	0	4	1	13	2	0	20	65%
5.	0	1	0	2	3	1	7	42,86%
6.	0	0	0	0	0	0	0	0%
False Detection	11	1	1	6	0	11	30	
Σ	31	106	56	81	10	16	300	

VII. CONCLUSION

In this work a detection and neural network based vehicle classification method was presented. The number of false detections was pretty big, so a filtering algorithm has been also developed. The filtering algorithm can exclude almost 60 percent of false detections.

Neural network training has been done with different number of hidden layer neurons, and different learning rates. Training results showed that the recognition rates are not usable in real-life applications, but the results are perspective. For better recognition rates, changes are needed. Increasing the number of training samples, dividing the vehicles into more classes, or dividing them by length and axle number could all lead to better results. Probably the most efficient modification could be to include the changes of waveforms in time.

ACKNOWLEDGMENT

The publication/presentation is supported by the European Union and co-funded by the European Social Fund. Project title: "Telemedicine-focused research activities on the field of Mathematics, Informatics and Medical sciences" Project number: TÁMOP-4.2.2.A-11/1/KONV-2012-0073.

The authors would like to thank companies “SELMA” Ltd. and “SELMA Electronic Corp” Ltd. for the technical resources and support.

REFERENCES

[1] A. Daubaras and M. Zilis, “Vehicle Detection based on Magneto-Resistive Magnetic Field Sensor”, Electronics and Electrical Engineering, Kaunas, Vol. 118, 2012, pp. 27-32.

[2] X. Deng, Z. Hu, P. Zhang and J. Guo: “Vehicle Class Composition Identification Based Mean Speed Estimation Algorithm using Single Magnetic Sensor”, Journal of Transportation Systems Engineering and Information Technology, ScienceDirect, 2010, pp 35-39.

[3] A. Haoui, R. Kavaler and P. Varaiya, “Wireless magnetic sensors for traffic surveillance”, *Transportation Research Part C: Emerging Technologies*, Vol. 16, ScienceDirect, 2008, pp. 294-306.

[4] S.-Y. Cheung and P. Varayra, “Traffic Surveillance by wireless sensor networks:Final Report”, California PATH Research Report, 2007.

[5] Isaksson M., “Vehicle Detection using Anisotropic Magnetoresistors”, Thesis For The Degree Of Master In Engineering Physics, Chalmers University Of Technology, 2007.

[6] E. Jouseau and B. Dorizzi, “Neural networks and fuzzy data fusion. Application to an on-line and real time vehicle detection system”, *Pattern Recognition Letters*, Vol. 20, Elsevier, 1999, pp. 97-107.

[7] W. Zhang, G.-Z. Tan, H.-M. Shi and M.-W. Lin “A Distributed Threshold Algorithm for Vehicle Classification Based on Binary Proximity Sensors and Intelligent Neuron Classifier”, Journal of Information Science and Engineering, Vol. 26, 2010, pp. 769-783.

[8] M.J. Caruso and L.S. Withanawasam, “Vehicle Detection and Compass Applications using AMR Magnetic Sensors”, Honeywell Inc., 2007.

[9] G. Zhang, Y. Wang and H. Wei, “An Artificial Neural Network Method for Length-based Vehicle Classification Using Single-Loop Outputs”, Journal Of The Transportation Research Board, Vol. 1945, 2007, pp. 100-108.

[10] C. Sun, “An Investigation in the Use of Inductive Loop Signature for Vehicle Classification”, California PATH Research Report, 2000, pp. 499–512.

[11] S.-Y. Cheung, S.E. Coleri and P. Varayra, “Traffic Surveillance by Wireless Magnetic Sensors”, ITS World Congress, 2005.

[12] S.-Y. Cheung, S. Coleri, B. Dundar, S. Ganesh, C.-W.Tan and P. Varayra, “Traffic Measurement and Vehicle Classification with a Single Magnetic Sensor”, 84th Annual meeting of the Transportation Research Board, 2005, pp. 173-181.

[13] This work was supported by TÁMOP-4.2.2.A-11/1/KONV-2012-0073.

[14] Liu H., Jeng S.-T., Tok J.C.A. and Ritchie S.G.: Commercial Vehicle Classification using Vehicle Signature Data, 88th Annual meeting of the Transportation Research Board, 2009.