

UNIVERSITY OF SZEGED
FACULTY OF SCIENCE AND INFORMATICS
DEPARTMENT OF TECHNICAL INFORMATICS

Doctoral School of Computer Science

**New methods in the application of inertial and
magnetic sensors in online pattern recognition
problems**

Ph.D. Thesis

Péter Sarcevic

Supervisor

Dr. Szilveszter Pletl

College professor

SZEGED

2018

1 Introduction

Due to the advances in technology, accelerometers, gyroscopes, and magnetometers became widely used in a large variety of applications.

Accelerometers are motion sensors, which measure linear acceleration in one or more axes. The velocity of an object can be calculated by integrating the object's acceleration over time. By integrating again, the position can be determined.

Gyroscopes are rotation sensors used to measure angular velocity and are sensitive around a single axis or multiple axes (usually two or three). Orientation can be calculated by integrating once the measured angular velocity.

Magnetometers are passive sensors that measure the strength of the Earth's magnetic field at a given point. These sensors are usually used as compasses; thus, they are capable of estimating heading direction [1].

These sensors are most commonly used in Inertial Navigation Systems (INS) to obtain information about the position, the orientation, and the speed of a moving object [2]. All three sensor types can be also found in consumer electronics, such as smartphones, tablets, or smartwatches, due to their small size, light weight, low cost, and low power consumption.

Inertial and magnetic sensors are widely used in different pattern recognition applications, such as human motion recognition [3], gesture recognition [4], fall detection and classification [5], vibration analysis [6], etc. Often, they are utilized in these applications in the form of Wireless Sensor Networks (WSN). The necessary computation in these systems can be done offline or online. An external unit is applied in the case of offline computation, while in online applications, the algorithms run on the used embedded systems. The design of online applications is a challenging task, since they require appropriate usage of energy, memory, and processing.

In this thesis, research results achieved in two pattern recognition applications of inertial and magnetic sensors are presented. The developed algorithms for both applications are online, thus, they are easily implementable on the used microcontroller-based embedded systems.

The first application applies accelerometers, gyroscopes, and magnetometers to classify different arm and body movements. Two sensor boards are used, which are attached to the wrists of the subjects. The proposed algorithms are based on features extracted from the signals generated by the changes in the position and the orientation of the sensors.

The second application utilizes a single magnetometer-based system to classify vehicles into multiple classes. The sensor is in stationary position, and the distortions in the measured magnetic field caused by passing vehicles are used during the extraction of features.

2 Movement recognition using wearable inertial and magnetic sensors

The analysis and real-time monitoring of human body motion is a widely-studied field of industrial, entertainment, health, and medical applications [7].

The sensor devices used in body sensor networks must be designed with the aim of providing the highest degree of mobility for the patients. They must be small, lightweight, and wireless wearable units.

The used prototype system consists of an IRIS WSN mote, and a nine degrees of freedom (9DoF) digital sensor board connected to it [8]. The IRIS mote is equipped with an Atmel ATmega 1281L 8-bit microcontroller, and an RF231 IEEE 802.15.4 compatible radio transceiver. The connected 9DoF sensor board is made up of an ADXL345 tri-axial MEMS accelerometer, an ITG3200 tri-axial MEMS gyroscope, and an HMC5883L tri-axial magnetoresistive technology-based magnetometer. A TinyOS-based driver was developed and implemented to configure the sensors and cyclically read the measurement data. The data are read from the sensors via the I2C interface, and sent via wireless communication to a BaseStation mote, which uses serial communication to forward the data to a PC.

Eleven activities were defined in order to recognize specific arm movements in stationary positions and also during the movement of the body: “standing without movement of the arms”, “sitting with the arms resting on a table”, “walking”, “turning around in one place”, “jogging”, “raising and lowering the left arm during standing”, “raising and lowering the right arm during standing”, “raising and lowering both arms during standing”, “raising and lowering the left arm during walking”, “raising and lowering the right arm during walking”, “raising and lowering both arms during walking”. Data were collected with the help of nine subjects for all activities. The IRIS motes with the attached 9DoF sensor motes were mounted on each wrist of the subjects. The data were recorded in fixed-length sessions of 20s for all activities using 125Hz sampling frequency, which means 2500 measurements per sensor.

Online movement classification algorithm

In the proposed algorithm, the classification is performed in four main stages. The software architecture with the four stages can be seen in Figure 1. The proposed algorithm assumes the transmission of the vector of the extracted features from one mote to the other, and the rest of the algorithm should be implemented in the microcontroller of the receiving device [9-10].

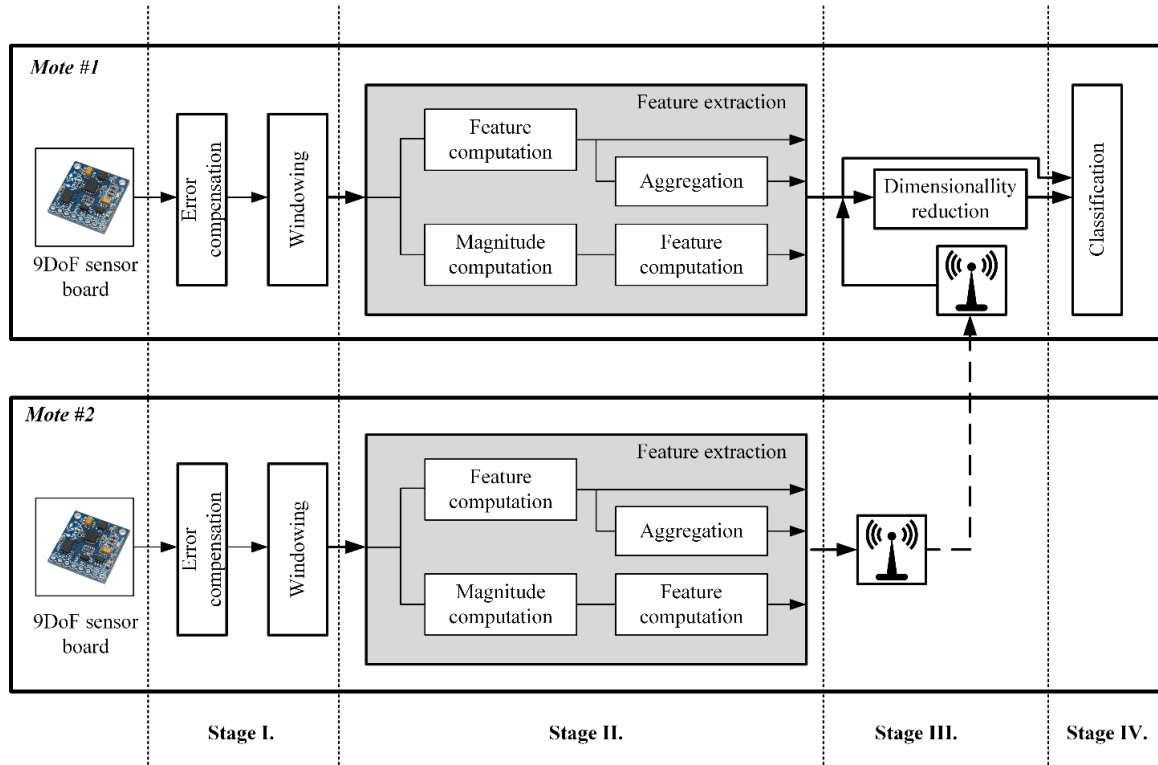


Figure 1: Software architecture.

Due to high error rates caused by structural errors of the sensors, the raw measurements are compensated in the preprocessing phase. The calibration parameters were obtained using an offline, evolutionary algorithm-based method, which applies multi-position measurements during the computation [11].

The extraction of feature values is performed in fixed-size segments, which are shifted with constant sizes.

The used features were chosen by their memory usage, required computation, and possible quantity of information. Due to easy implementation and low memory usage, only time-domain analysis was performed on the signals. The following time-domain features (TDFs) were chosen for this research: Mean Absolute Value (MAV), Willison Amplitude (WAMP), Number of Zero Crossings (NZC), Number of Slope Sign Changes (NSSC), Maximal (MAX) and Minimal (MIN) value, Root Mean Square (RMS), and Waveform Length (WL).

The used input vectors were generated and tested with the use of two TDF calculation modes: features extracted separately for the X, Y, and Z axes of the sensors (SEP), and vector magnitude-based extraction (VL), where the changes in the vector length are used for the computation of the TDFs.

The usage of the separately extracted features for the three sensor axes can result in a very high number of features, which can increase the complexity of the classification algorithm. Also, it can have a negative effect on the recognition efficiency if the subjects do not fix the units correctly to their wrists. A possible solution to both previous problems can be the

aggregation (AGG) of the separately computed features. As expressed in (1), this can be done by calculating a linear combination of the feature values computed for each axis for a specific feature type.

$$feat_{AGG} = w_X \cdot feat_X + w_Y \cdot feat_Y + w_Z \cdot feat_Z, \quad (1)$$

where $feat_{AGG}$ is the aggregated feature value, $feat_X$, $feat_Y$, and $feat_Z$ are the extracted features for each axis, and w_X , w_Y , and w_Z are the corresponding weights.

The Linear Discriminant Analysis (LDA) method was used to perform dimensionality reduction on the datasets. The result of this method is a matrix of parameters, which must be multiplied with the feature vector to get the inputs of the classifier. Thus, it requires only multiplication and summation for its implementation.

Seven possibly applicable classification methods were chosen and tested: Nearest Centroid Classifier (NCC), Multi-Layer Perceptron (MLP), Naïve Bayes Classifier (NBC), “one-versus-one” (OvO) and “one-versus-all” (OvA) Support Vector Machines (SVM), k-Nearest Neighbor (k-NN), Classification Tree (CT).

Altogether 340 datasets were constructed using different combinations of used sensor types, TDF calculation modes, processing window sizes, and sampling frequencies. All defined processing window sizes were in millisecond domain. The following window width and shift pairs were tested: 80ms width and 40ms shift; 200ms width and 40ms shift; 400ms width and 80ms shift; 800ms width and 80ms shift. The three sensor types were tested separately, in pairs, and altogether, to investigate the impact of the sensors on recognition efficiency, and to prevent unnecessary usage of memory and hardware resources. Datasets were generated using five sampling frequencies: 25Hz, 50Hz, 75Hz, 100Hz, and 125Hz. The data for the four lower frequencies were obtained by downsampling the measurement data collected with 125Hz sampling frequency.

Data from five of the nine subjects were used for the training of the classifiers, while the data from the remaining four subjects were tested as unknown inputs for the validation of the trained classifiers.

All seven classification techniques were tested for all datasets with and without dimension reduction to examine its effect on recognition efficiency, memory consumption, and training time. No results could be achieved using the NBC without LDA, since some classes have features with zero variance. The results showed that the highest efficiency can be achieved using the MLP classifier, but the use of the LDA-MLP is also reasonable due to the slightly lower efficiency, lower memory requirements in the case of high feature numbers, and significantly lower training time. The CT can effectively classify the training data, but its performance is significantly lower for the unknown samples. The very popular k-NN and SVM-based methods showed to be unsuitable for use in this application due to their low efficiencies and high hardware requirements.

The best results for the 17 different combinations in the four different processing window widths can be seen in Figure 2.

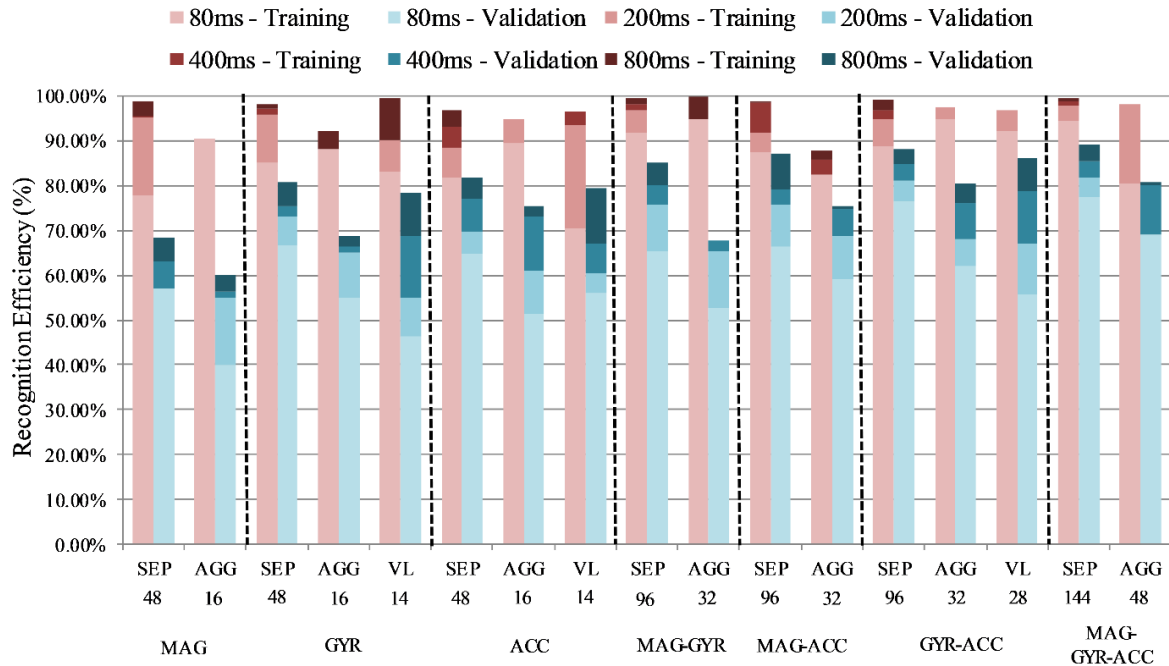


Figure 2: Achieved classification efficiencies on training and validation data using different processing window sizes. The horizontal axes show the feature extraction mode in the first row, the required feature numbers in the second row, and the used sensor types in the third row.

The results show that the recognition rates achieved, using only simple time-domain features, are not affected significantly by the sampling rate, and only slight improvements can be noticed when it is increased. The tested millisecond range processing windows prove to be usable, since above 77% percent efficiency can be reached on unknown data even with the smallest, 80ms, window width, while almost 90% can be achieved with the 800ms window size. It can be concluded from the achieved efficiencies, that the movements of different subjects show high correlation, since the training and validation datasets were constructed of data from different persons. The classification rates on training data can be almost 100%, which is also a very important factor if the application should be trained and used for one person.

The magnetic sensor itself provides very low, 40-67%, recognition rates, but it can significantly improve the performance of the gyroscope and the accelerometer if they are used together. The two inertial sensors alone can provide around 80% applying the largest processing window. The highest efficiencies were achieved when the data from the three sensor types were applied together, but the impact of the magnetometer is very small, since it only increases the recognition rates by 1-2%, while it largely increases the cost, the energy consumption, and the required feature number. For some datasets, adding the features extracted from the magnetometer data can even lower the classification efficiency.

The highest efficiencies were achieved when the separately computed features were used, but they require three-times more used features than the aggregation- and magnitude-based datasets. The inertial sensors can provide 86.17% using the VL-based extraction in the 800ms processing window, but for the smaller window sizes the proposed aggregation-based feature extraction provides higher classification rates.

Hierarchical-distributed movement classification algorithm

Since the previous online classification algorithm uses together the features extracted on both sensor motes in the classification process, its implementation requires high energy consuming radio communication for data transfer between the motes. It is reasonable to split the classification algorithm into a hierarchical approach to get a distributed network, so the motes can calculate their own movement classes [12-13]. Difference compared to the previous algorithm can be noticed only after the preprocessing part, since error compensation, windowing, and feature extraction are done identically. Classification is performed on both motes based on the computed feature vectors. Using the proposed hierarchical-distributed technique, only the computed movement class is needed to be transferred periodically from one mote to the other based on the value of the window shift. The determined classes are combined to get the movement of the entire body and arms. Besides that, using the proposed algorithm less data transfer is required via wireless communication, the classifiers have less input features and output classes, thus, it is more energy-efficient and easier to implement the algorithm on the motes. Adding new devices to the system would be also easier with this approach, since only the class combination stage on the receiving unit needs to be modified, and the size of the feature vector could also become too large in the case of the non-distributed approach.

In the distributed approach, some classes can be merged by the role of the arm in the given movement. This way the number of classes can be reduced to seven for one unit. Two different approaches were tested for the classification hierarchy. In the first approach, the movements are equally distributed, all of them are on the same level. In the second hierarchy, the movements are split into three levels. The classification algorithm uses five distributions to decide which element of the hierarchy matches the actual movement.

For classification, MLP neural networks were applied. Class combination is done on the unit which receives the other's movement class. The combination gives one of the eleven classes as the output, but it gives an unknown class if the classes are not a valid pair.

Altogether 68 datasets were constructed to examine the effect of the processing window size. The signals measured using the 125Hz sampling frequency were utilized during feature extraction.

The achieved results show, that the two hierarchies provide similar results, but the non-distributed approach gives much higher results. The average difference is around 25% compared to both hierarchical approaches for both training and validation data. The highest

classification efficiencies on validation data, 75.76% using the first and 72.05% applying the second hierarchy, can be achieved using the three sensor types together.

Comparison of time- and frequency-domain analysis

Since in the proposed algorithms only TDFs with low memory and computation requirements were applied, it is important to compare the capabilities of the two extraction modes on the collected measurement data [14]. The Standard Deviation (STD) was added to the previously used TDFs. Frequency Domain Features (FDFs) are computed using the amplitude spectrum of the obtained signals, thus, they require transformation to frequency-domain. The transformation was done using the Fast Fourier Transform (FFT). The applied FDFs were the following: Spectral Energy (ENE), Median Frequency (MDF), Mean Frequency (MNF), Mean Power (MNP), Peak Magnitude (PKM), Peak Frequency (PKF), and Variance of the Central Frequency (VCF). Three different processing window widths were applied: 64, 128, and 256 measurements. Using the applied 125Hz sampling frequency, this means nearly 0.5s, 1s, and 2s.

The obtained results show, that the TDFs provide higher recognition rates in most of the setups. The highest recognition rates for all three tested processing window widths were achieved using the raw data from the gyroscopes and accelerometers. The overall highest efficiencies were obtained using the largest window size, 91.74% using TDFs, and 88.51% utilizing FDFs.

3 Real-time vehicle classification using a single magnetometer

Automatic vehicle detection technologies play an important role in Intelligent Transportation Systems (ITS). These systems provide vehicle count and classification data, which are very important inputs for traffic modelling, pavement design, transportation planning, emission/pollution estimation, and traffic control. The most widely used technologies are inductive loops and Video Image Processing (VIP)-based systems, but magnetometer-based vehicle detectors have many advantages compared to these technologies.

Vehicle classification can be realized by using two sensor units, which allows the computation of vehicle speed and length, or by applying only one sensor unit, but these systems must rely only on magnetic signatures in the classification process [15].

A single measurement unit was installed for data acquisition. The detector was mounted into the surface of the pavement in the middle of the lane. The used hardware is an HMC5843-based unit, which was developed by Selma Ltd., Subotica, Serbia. The hardware is also equipped with an 8-bit microcontroller (Silicon Labs C8051F930), which can communicate with the sensor and external equipment. The developed vehicle detection algorithm was implemented on the installed measurement unit. The applied sampling frequency was 50Hz. The software of the device sent the measurement data and the value of the detection flag to a server. To validate the detections, and to determine the class for each recorded sample, a camera was installed beside the road. The data acquisition software saved camera images at every falling edge of the detection flag. Data acquisition was done for multiple months in various weather conditions, and altogether more than 30000 samples were collected.

Sensor calibration and vehicle detection algorithm

Vehicle detection is a very important part of the classification process, since it should follow the environmental changes [16], and act the same way for different vehicles. The developed detection algorithm, which is designed mainly for vehicle classification purposes, applies adaptive thresholds to follow these changes [17]. First, a calibration process is done, which finds the current offset values on each axis. It uses a calibration range size, which must be slightly larger than the peak-to-peak noise level. During calibration the highest and lowest measurement values are monitored in a measurement window for all three axes. If the calibration is successful, the upper and lower calibration and detection thresholds are calculated for all axes. This is done by equally stretching the range determined by the highest and lowest values to the width defined by the range sizes. Vehicle presence is declared if the measurement values exceed the detection threshold at both of the X and Z axes. The detection flag is cleared if the measurement values on both X and Z axes are between the calibration thresholds for a previously defined number of measurements. The algorithm should be suitable for detecting vehicles with trailers, so the used length should be calculated using the potential speed on the

location and the possible distance between the vehicle and the trailer. A recalibration is attempted always when vehicle presence is not declared, which enables the following of the environmental changes.

A one-hour test was done to test the efficiency of the algorithm. The overall efficiency of the detection algorithm is 94.15% [18]. The algorithm detects all vehicles passing above the sensor, only motorcycles can cause failures if they are not passing near to the detector. False detections cause the rest of the failures. These are generated by vehicles with high metallic content passing in the neighboring lane.

Vehicle classification algorithm

The used vehicle classification algorithm consists of three main parts: vehicle detection algorithm, feature extraction, and classification [19]. To perform real-time and online vehicle classification, all parts should be easily implementable on the microcontroller of the used hardware.

Features are extracted in the detection window. The chosen feature types are based on only time domain analysis, because they need little computation, and they also do not require the storage of all measurements in the window. Feature types were chosen by their potential ability to detect vehicle axles and overall metallic content. It is also important that the features should be immune to speed variations. The used feature types were the following: MAX, MIN, Place of the highest and lowest values, range changes, number of local maxima and minima, MAV, Mean Value (MV), NSSC, NZC, Average waveform length (AWL), RMS, and WAMP.

Various extraction modes were tested, which can possibly increase efficiency and/or decrease the required feature number. Feature extraction is done using raw sensor measurements and aggregated data, which are computed using the raw measurement data. The following extraction modes were applied: measurement axes, absolute values, magnitudes from the origin, angles, and magnitude from the calibration point.

Taking into account the capabilities of magnetic sensors, nine possibly separable vehicle classes were defined: motorcycle, car, car with trailer, van and mini bus, truck, truck with trailer, tractor trailer, bus, articulated bus.

Three-layer MLP neural networks were applied for classification in the proposed algorithm, and 130 samples were utilized for each class.

Different datasets based on different feature extraction modes and combinations of used sensor axes were tested to minimize the number of used features and the possible cost of the system. Altogether 18 different combinations were defined. The effect of the detection length (DL) in the feature set was also examined.

Table 1 summarizes the required feature numbers and average recognition rates on training and validation samples for each of the 36 datasets.

Table 1: Feature numbers and average recognition efficiencies on training and validation data for different setups.

Used axes					
Feature number					
Efficiency on training data (%)			Efficiency on validation data (%)		
X, Y, Z 42 82.11 67.69	X, Y, Z, DL 43 81.50 66.58	X 14 78.83 71.33	X, DL 15 80.11 72.22	Z 14 77.92 71.24	Z, DL 15 81.36 72.36
X, Z 28 81.61 70.13	X, Z, DL 29 81.97 71.64	$X_{abs}, Y_{abs}, Z_{abs}$ 27 81.00 71.20	$X_{abs}, Y_{abs}, Z_{abs}, DL$ 28 81.81 69.73	X_{abs} 9 73.86 68.18	X_{abs}, DL 10 76.14 70.04
Z_{abs} 9 74.31 69.07	Z_{abs}, DL 10 75.56 69.82	X_{abs}, Z_{abs} 18 78.97 69.96	X_{abs}, Z_{abs}, DL 19 80.08 71.38	$XY_O, XZ_O, YZ_O,$ XYZ_O 56 82.20 70.18	$XY_O, XZ_O, YZ_O,$ XYZ_O, DL 57 82.81 68.18
XZ_O 14 80.75 73.73	XZ_O, DL 15 80.14 74.44	XY_O, XZ_O, YZ_O 42 81.03 69.82	$XY_O, XZ_O, YZ_O,$ DL 43 84.06 71.51	XYZ_O 14 77.97 72.76	XYZ_O, DL 15 81.39 74.67
XY_A, XZ_A, YZ_A 42 79.00 66.53	XY_A, XZ_A, YZ_A, DL 43 79.97 65.78	XZ_A 14 80.92 71.20	XZ_A, DL 15 79.69 71.87	$XY_C, XZ_C, YZ_C,$ XYZ_C 36 79.08 66.36	$XY_C, XZ_C, YZ_C,$ XYZ_C, DL 37 81.75 69.11
XZ_C 9 72.78 68.49	XZ_C, DL 10 76.53 70.53	XY_C, XZ_C, YZ_C 27 80.83 68.36	$XY_C, XZ_C, YZ_C,$ DL 28 79.28 68.27	XYZ_C 9 73.61 68.93	XYZ_C, DL 10 75.69 71.38

The obtained results show, that using aggregated data in the form of angles or magnitudes, can even increase the classification efficiencies besides decreasing the number of inputs. The highest recognition rates on validation data were achieved using magnitudes computed from the origin in the XZ-plane, 74.67% with, and 73.73% without applying the detection length. Using only the X or the Z axis can provide 71-72%, which shows that even a single axis sensor can effectively classify vehicles into multiple classes. The length of the detection in average increases the classification efficiencies only by 1.28% \pm 1.63% for training data, and 0.80% \pm 1.37% for unknown samples, which shows that the used features can effectively extract the information from the waveforms. The necessary feature number in the best sets was 14-15, while convergence on unknown samples can be noticed with 10-15 neurons.

Rule-based false detection filtering algorithm

The overall efficiency of the classification algorithm can be largely affected by the false detections, thus, it is important to minimize their number. The proposed false detection filtering

method is based on different rules, which are constructed using different data types and relations [18, 20].

Finding optimal parameters and relations was tested with different data types. Beside the measurement values on the three sensor axes, aggregated data computed using the measurement values on the three axes were also tested. The axis pointing to the neighboring lane should carry the most information for false detection filtering, but other sensor axes were also considered. The following data types were utilized: measurement axes, magnitudes from the origin, and angles.

The following value types were determined in the detection window for all data types: highest value, lowest value, and highest value in the case of absolute values of the measurements.

Since a vehicle with high metallic content passing in the neighboring lane can generate larger differences in the signals than a vehicle with lower metallic content passing above the sensor, it is reasonable to test ratios computed using different data. Ratios were calculated for all combinations in different data types in the case of all value types.

Genetic algorithms were used for the optimization of parameter values. The mistakes in the case of correct detections is allowed, but the rate of both correctly determined good and false detections are used in the fitness function. The rates are weighted, since it is less important to detect false detections than to lose correct ones. The used fitness function is the following:

$$fitval = w_c - w_c \cdot n_c / N_c + w_f - w_f \cdot n_f / N_f \quad (2)$$

where *fitval* is the fitness value, n_c is the number of found correct detections, N_c is the number of all correct detections, n_f is the number of found false detections, N_f is the number of all false detections, w_c is the weight used at correct detections, and w_f is the weight used at false detections.

To find the parameter types which can be effectively used for the filtering of false detections, first all combinations of data types and value types were tested separately. After finding the most useful parameters, it is important to combine these parameters in more complex rules for more accurate results. The rules were combined in two different modes, since both the “AND” and “OR” logical operators between the separate rules were tested during the examination of the samples.

The achieved results show that 97% of false detections can be filtered with losing only nearly 0.3% of good detections. The highest results can be obtained using computed magnitudes, and slightly lower rates can be achieved using measurements on sensor axes. A single parameter can be also sufficient for the filtering of the false detections. It can be also noticed that despite that the axis pointing to neighboring lane (Y) should carry most information for the filtering of false detections, the other two axes provide more reliable parameters.

4 Summary of the thesis

The relation between the theses and the corresponding publications can be seen in Table 2.

Table 2: Theses and corresponding publications.

Publication	Thesis							Type
	1.1	1.2	1.3	2.1	2.2	2.3	2.4	
[8]	•	•						conference paper
[9]		•						conference paper
[10]		•						journal paper
[12]			•					conference paper
[13]			•					journal paper
[14]		•						conference paper
[17]				•	•			book chapter
[18]					•	•	•	conference paper
[19]						•		journal paper
[20]							•	conference paper

The results of the thesis can be summarized as follows:

Thesis Group 1. I developed novel online algorithms for the classification of human body and arm movements using wrist-mounted accelerometers, gyroscopes and magnetic sensors. The algorithms are implementable on resource constrained embedded systems.

Thesis 1.1. I developed a prototype measurement system to record human movements and collected measurement data for training and validation of the classifiers. I used two nine degrees of freedom (9DoF) sensor boards mounted on WSN motes, which were attached to the wrists of the subjects.

Thesis 1.2. I developed a new online movement classification algorithm where the features extracted on multiple units are used together in the classification stage.

I showed that the use of only time-domain features (TDFs) with low memory requirements is sufficient to obtain high classification efficiencies in movement recognition systems using inertial and magnetic sensors.

I showed that there is a high correlation between the movements of different subjects, since the classifiers provided very high recognition rates on measurements of subjects, which were not used during the training.

I showed that millisecond domain in processing window widths is sufficient for efficient recognition of human body and arm movements.

I proposed an aggregation-based feature reduction method for sensors with multiple measurement axes, which results in three-times less features than when the features computed for each sensor axis are used as inputs of the classifiers. The aggregation is done by calculating

a linear combination of the feature values computed for each axis for a specific feature type. The aggregation method can provide useful classification efficiencies. This reduction method can also help the system to be less sensitive to differences in orientations of the sensors on the arms.

I showed that adding a new sensor type can in some cases only slightly increase or even decrease the overall recognition rate of the system. With testing the possible sensor types in different combinations, an optimal configuration can be found, which can lower the overall cost and power consumption of the system.

I showed that the Multi-Layer Perceptron (MLP) neural networks are the optimal solution for online algorithms in similar applications, and that some popular methods (SVM, k-NN) are not suitable for online use in resource constrained embedded systems.

I showed that the use of LDA-based dimension reduction on the feature sets can improve recognition efficiency, training time, and memory requirements for implementation of different classifiers.

I showed that using only simple TDFs even higher recognition rates can be achieved for some setups than when only features extracted using frequency-domain analysis are applied.

I showed that the proposed aggregation-based feature reduction can be also effectively applied in the case of frequency-domain features (FDFs).

Thesis 1.3. I developed a hierarchical-distributed algorithm for online human movement classification, where the multiple units determine their own movement classes, and one unit combines the resulted classes to determine the movement class of the entire body. This algorithm requires less computation because of the smaller classifiers and less wireless communication, which improves energy efficiency. The addition of new units to the system would be also easier using this approach. I applied two different movement hierarchies, which provide similar results.

Thesis Group 2. I developed a new real-time vehicle classification system based on a single magnetic sensor mounted into the surface of the pavement.

Thesis 2.1. I developed a prototype measurement system with a single magnetometer-based unit installed into the pavement surface. I also used a camera mounted on the side of the road to validate the detections and to determine the class of the samples. I collected a very large database of samples using the measurement system.

Thesis 2.2. I developed a vehicle detection algorithm based on adaptive thresholds. The algorithm can deal with the environmental changes and is mainly designed for vehicle classification purposes. The algorithm estimates the offsets on each sensor axis using a calibration method after the unit is turned on, and follows the environmental changes by continuous recalibration. The algorithm is also capable of detecting vehicles with trailers.

Thesis 2.3. I developed a new online vehicle classification algorithm, which can classify vehicles into nine vehicle classes using MLP neural networks. The algorithm uses TDFs, which can potentially detect vehicle axles and overall metallic content, are immune to speed variations, and require low memory usage.

I showed that a single magnetometer-based system is sufficient for classifying vehicles into nine vehicle classes.

I showed that using aggregated data in the form of angles or magnitudes, can even increase the classification efficiencies besides decreasing the number of inputs.

I showed that adding the detection length to the feature set, which is a speed-dependent feature, only slightly increases the classification efficiencies. This shows that the used features can effectively extract the information from the waveforms.

Thesis 2.4. I developed a rule-based algorithm for the filtering of false detections caused by vehicles with high metallic content passing in the neighboring lane. This can further increase the overall efficiency of the proposed vehicle classification system.

I showed that using proper rules, a very high percentage of false detections can be filtered with losing only a few good detections.

I showed that despite that the axis pointing to neighboring lane should carry most information for the filtering of false detections, the other two axes provide more reliable parameters.

I showed that aggregated data in the form of magnitudes can provide higher filtering rates than parameters extracted from the signals on the sensor axes.

References

- [1] W. Li, and J. Wang, Magnetic sensors for navigation applications: an overview. *The Journal of Navigation* 67(2): 263-275, 2014.
- [2] H. Zhao, and Z. Wang, Motion measurement using inertial sensors, ultrasonic sensors, and magnetometers with extended Kalman filter for data fusion. *IEEE Sensors Journal* 12(5): 943-953, 2012.
- [3] K. Altun, B. Barshan, and O. Tuncel, Comparative study on classifying human activities with miniature inertial and magnetic sensors. *Pattern Recognition* 43(10): 3605-3620, 2010.
- [4] K. Liu, C. Chen, R. Jafari, and N. Kehtarnavaz, Fusion of inertial and depth sensor data for robust hand gesture recognition. *IEEE Sensors Journal* 14(6): 1898-1903, 2014.
- [5] P. Tsinganos, and A. Skodras, On the comparison of wearable sensor data fusion to a single sensor machine learning technique in fall detection, *Sensors* 18(2): 592, 2018.
- [6] C. A. Brooks, and K. Iagnemma, Vibration-based terrain classification for planetary exploration rovers. *IEEE Transactions on Robotics* 21(6): 1185-1191, 2005.
- [7] M. Cornacchia, K. Ozcan, Y. Zheng, and S. Velipasalar, A survey on activity detection and classification using wearable sensors. *IEEE Sensors Journal* 17(2): 386-403, 2017.
- [8] P. Sarcevic, Z. Kincses, and S. Pletl, Wireless sensor network based movement classification using wrist-mounted 9DOF sensor boards. In: *Proc. of IEEE International Symposium on Computational Intelligence and Informatics (CINTI)*, Budapest, Hungary, pp. 85-90, 2014.
- [9] P. Sarcevic, Z. Kincses, and S. Pletl, Comparison of different classifiers in movement recognition using WSN-based wrist-mounted sensors. In: *Proc. of IEEE Sensors Applications Symposium (SAS)*, Zadar, Croatia, pp. 446-451, 2015.
- [10] P. Sarcevic, Z. Kincses, and S. Pletl, Online human movement classification using wrist-worn wireless sensors. *Journal of Ambient Intelligence and Humanized Computing*, 2017. <https://doi.org/10.1007/s12652-017-0606-1>
- [11] P. Sarcevic, S. Pletl, and Z. Kincses, Evolutionary algorithm based 9DOF sensor board calibration. In: *Proc. of IEEE International Symposium on Intelligent Systems and Informatics (SISY)*, Subotica, Serbia, pp. 187-192, 2014.
- [12] P. Sarcevic, Z. Kincses, S. Pletl, and L. Schaffer, Distributed movement recognition algorithm based on wrist-mounted wireless sensor motes. In: *Proc. of European*

- Wireless (EW)*, Budapest, Hungary, pp. 130-135, 2015.
- [13] P. Sarcevic, L. Schaffer, Z. Kincses, and S. Pletl, Hierarchical-distributed approach to movement classification using wrist-mounted wireless inertial and magnetic sensors. *Infocommunications Journal* 7(3): 33-41, 2015.
 - [14] P. Sarcevic, Z. Kincses, and S. Pletl, Comparison of time- and frequency-domain features for movement classification using data from wrist-worn sensors. In: *Proc. of IEEE International Symposium on Intelligent Systems and Informatics (SISY)*, Subotica, Serbia, pp. 261-265, 2017.
 - [15] H. Tafish, W. Balid, and H. Refai, Cost effective vehicle classification using a single wireless magnetometer. In: *Proc. of International Wireless Communications and Mobile Computing Conference (IWCMC)*, Paphos, Cyprus, pp. 194-199, 2016.
 - [16] S. Cheung, and P. Varaiya, *Traffic surveillance by wireless sensor networks: Final report*. University of California, Berkeley, 2007.
 - [17] P. Sarcevic, Vehicle classification using neural networks with a single magnetic detector. *Issues and Challenges of Intelligent Systems and Computational Intelligence, Studies in Computational Intelligence*, vol. 530, pp. 103-115, 2014.
 - [18] P. Sarcevic, and S. Pletl, Vehicle classification and false detection filtering using a single magnetic detector based intelligent sensor. In: *Proc. of International Conference on Information Society and Technology (ICIST)*, Kopaonik, Serbia, pp. 144-149, 2014.
 - [19] P. Sarcevic, and S. Pletl, Real-time vehicle classification system using a single magnetometer. *Acta Polytechnica Hungarica* (submitted).
 - [20] P. Sarcevic, and S. Pletl, False detection filtering method for magnetic sensor-based vehicle detection systems. In: *Proc. of IEEE International Symposium on Intelligent Systems and Informatics (SISY)*, Subotica, Serbia, pp. 277-281, 2018.