

Research Article

Road Traffic Monitoring System Based on Mobile Devices and Bluetooth Low Energy Beacons

Marcin Lewandowski,¹ Bartłomiej Płaczek ¹, Marcin Bernas ², and Piotr Szymała³

¹*Institute of Computer Science, University of Silesia, Sosnowiec, Poland*

²*Department of Computer Science and Automatics, University of Bielsko-Biala, Bielsko-Biala, Poland*

³*Institute of Innovative Technologies EMAG, Katowice, Poland*

Correspondence should be addressed to Bartłomiej Płaczek; placzek.bartlomiej@gmail.com

Received 28 February 2018; Revised 25 June 2018; Accepted 11 July 2018; Published 17 July 2018

Academic Editor: Carlo Giannelli

Copyright © 2018 Marcin Lewandowski et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The paper proposes a method, which utilizes mobile devices (smartphones) and Bluetooth beacons, to detect passing vehicles and recognize their classes. The traffic monitoring tasks are performed by analyzing strength of radio signal received by mobile devices from beacons that are placed on opposite sides of a road. This approach is suitable for crowd sourcing applications aimed at reducing travel time, congestion, and emissions. Advantages of the introduced method were demonstrated during experimental evaluation in real-traffic conditions. Results of the experimental evaluation confirm that the proposed solution is effective in detecting three classes of vehicles (personal cars, semitrucks, and trucks). Extensive experiments were conducted to test different classification approaches and data aggregation methods. In comparison with state-of-the-art RSSI-based vehicle detection methods, higher accuracy was achieved by introducing a dedicated ensemble of random forest classifiers with majority voting.

1. Introduction

Road traffic is a complex phenomenon, where various entities (pedestrians, cars, trucks, busses, tramps, bicycles, etc.) interact one each other, when using common infrastructure. The traffic management and control, due to infrastructure constraints and rising number of vehicles, is a complex task and requires application of dedicated algorithms together with precise traffic data (both historical and current) [1]. The information about number of vehicles and their types is helpful in reducing travel times and emissions [2]. Precise traffic data allows us not only to increase effectiveness of traffic control, but also to adapt management policy to changing conditions and predict infrastructure bottlenecks [3].

The precise traffic data can be provided by traffic monitoring systems that are usually integrated with road infrastructure. Such systems allow detecting and classifying the vehicles in selected areas by using data from sensors (inductive loops, video-detectors, magnetometers, etc.) [4]. A major drawback of the solutions integrated with infrastructure is a low flexibility and significant maintenance cost. To overcome these

drawbacks, applications of new technologies (e.g., wireless sensor networks) in traffic monitoring are considered [5]. Such solutions can facilitate installation and reconfiguration of the system. However, the cost is still significant.

Thus, in this paper an alternative method was proposed, which was inspired by the crowd sourcing approaches and utilizes iBeacon techniques for vehicle detection and classification. Crowd sourcing [6] is a distributed model, in which a crowd solves or helps to solve a complex problem. Crowd sourcing utilizes mobile workforce and unique features, which could be found in smartphones. Smartphones offer a great platform for extending existing applications due to multisensing capabilities: geolocation, audio, and visual sensors. They could be used to provide precise data about current traffic at given location. In contrast to the approximation models proposed in [7], where mobile device is situated inside a vehicle, this paper proposes a new system with mobile devices (smartphones) and beacons situated by the road. In order to detect vehicles, the proposed system measures signal strength of frames received from Bluetooth beacons.

According to the proposed traffic monitoring approach, a wireless network is composed of smartphones and battery-powered beacons. The beacons and the smartphone devices are placed on opposite sides of a road. The smartphones register broadcasted beacon frames and collect values of received signal strength indicator (RSSI). The proposed method utilizes RSSI data collected by mobile devices (e.g., smartphones) to recognize passing vehicles in a given area. These tasks are performed by using a proposed ensemble of classifiers, which significantly increases the vehicle detection and classification accuracy.

The novel aspects of this study include (1) proposal of a vehicle detection and classification system, which is suitable for the crowd sourcing applications; (2) design of classifier ensemble that enables utilization of RSSI data for accurate vehicle detection and classification; (3) verification of the new Bluetooth-based traffic monitoring system in real-traffic conditions.

The paper is organized as follows. Section 2 includes a survey of related literature. Details of the proposed hybrid traffic monitoring system are discussed in Section 3. Experiments and their results are described in Section 4. Finally, conclusions are given in Section 5.

2. Related Works and Contribution

Smartphones become the round-the-clock interface between user and the environment, which integrates the Internet network (via WiFi, 2G/3G/4G/5G) with local-area networks (e.g., Bluetooth, new generation NFC, or Portable WiFi, which allows the smartphone to act as a router and share the cellular connection with nearby devices) [6]. It is worth noting that each of these communication standards is characterized by different energy consumption and data transfer parameters [8]. Smartphone devices possess powerful computational capabilities and are equipped with various functional built-in sensors [9] that have enabled the development of mobile sensing technologies [10–14]. Among them, crowd sensing [12] plays important role due to the possibility of collecting useful data. The crowd sensing approach utilizes large amounts of participants to monitor the surrounding environment by means of various sensors: accelerometer, gyroscope, compass, microphone, camera, GPS, and wireless network interfaces.

The mobile sensing technologies were used for the development of noise detection [14], social behavior monitoring [15], health monitoring of disabled patients [16], and indoor localization [17, 18]. Another example is an accurate and energy-efficient smartphone-based traffic lane detection system for vehicles, which can detect different lane-level landmarks with accuracy above 90% [19]. Several solutions were also proposed for road traffic monitoring [20, 21]. These solutions provide the GPS localization data for vehicle tracking. They require the mobile device to be present in vehicle; thus not all vehicles can be tracked in this way.

In this paper a method is proposed, which allows the smartphones placed in road surrounding (e.g., on sidewalks in pedestrians' pockets) to be used for traffic monitoring. According to the introduced method, vehicle detection and

classification is performed by analyzing strength of radio signal received from Bluetooth beacons.

Up to date, several efforts have been made to explore the possibility of vehicles detection and localization via channel state information (CSI) [22], received signal strength indicator (RSSI) [23, 24], link quality indicator (LQI), and packet loss rate [25].

A method, which uses wireless transmission to detect road traffic congestion, was proposed in [25]. This method requires a pair of wireless transmitter and receiver. The transmitter continuously sends packets. The receiver, which is placed on opposite side of a road, evaluates RSSI, LQI, and packet loss metrics. It was shown that these metrics enable recognition between free-flow and congested traffic states with high accuracy. The method was implemented and tested with use of ZigBee nodes.

Similar ZigBee network was adapted in [24] for vehicles detection. The experimental results presented in that work confirm that a vehicle passing between the network nodes causes a drop of RSSI value. It was also observed that the gradient of RSSI drop depends on the vehicle speed.

In [26] a method was introduced for vehicle detection and speed estimation, which is based on RSSI analysis in network composed of two WiFi access points and two WiFi-equipped laptops. Mean value and variance of RSSI measurements were used to discriminate between three states: empty road, stopped vehicle, and moving vehicle. The experimental results reported in [26] show that variance of RSSI decreases with increasing speed of vehicle. This dependency was used for speed estimation.

Another WiFi-based traffic monitoring system was presented in [22]. This system utilizes single access point and one laptop to provide functionalities of vehicle detection, classification, lane identification, and speed estimation. According to that approach, CSI patterns in WiFi network are captured and analyzed to perform the traffic monitoring tasks. The CSI characterizes signal strengths and phases of separate WiFi subcarriers.

In [23] a radio-based approach for vehicle detection and classification was introduced, which combines ray tracing simulations, machine learning, and RSSI measurements. The authors have suggested that different types of vehicles have specific RSSI fingerprints. This fact was used to perform a machine-based vehicle classification. The RSSI values were analyzed in a wireless network of three transmitting and three receiving units, which were positioned on opposite sides of a road. The six wireless units were mounted on delineator posts and equipped with directional antennas. It was demonstrated that such system is able to detect vehicles and categorize them into two classes (passenger car and truck). It was also demonstrated that traffic lanes in a two-lane road have different distributions of CSI data. This fact was utilized to identify in which lane a vehicle is detected.

The wireless networks have been also used for detection of parked vehicles. In order to detect the parked vehicles, the transmitting nodes are placed on parking space and the receiving nodes are installed at a high location. When a vehicle is parked over the transmitting node, a decrease of the RSSI value is registered. Thus, the vehicles can be

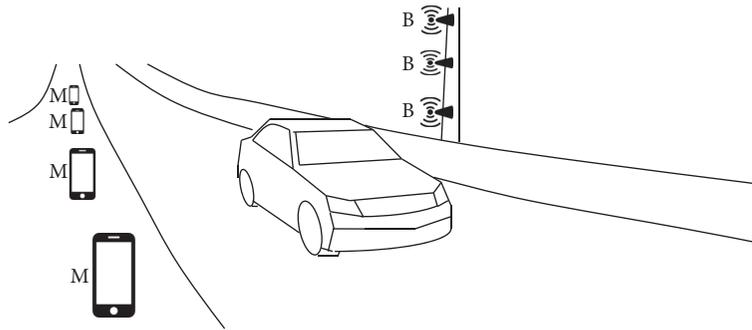


FIGURE 1: Placement of mobile devices (M) and beacons (B).

easily detected based on simple RSSI analysis. Different systems of this type were implemented with use of CC1101 wireless communication modules [27] and XBee motes [28].

The above-discussed methods from the literature are not suitable for the crowd sourcing applications, as they require energy-expensive data transfers (WiFi) or specialized hardware (ZigBee modules, directional antennas). The new approach proposed in this paper utilizes the Bluetooth low energy (BLE) communication, which is commonly available in smartphones. According to the introduced approach, BLE beacons are used with iBeacon protocol [29] to broadcast data frames. The beacon frames are registered by smartphones that collect the RSSI measurements, aggregate them, and send to a server for further analysis. It should be noted here that the BLE beacons are cheap battery-powered devices that can work for a long time (years) without battery replacement or charging. Moreover, the use of BLE communication significantly extends the lifetime of smartphone battery in comparison to WiFi transmission [30]. Nevertheless, beacon discovery has a significant impact on smartphone battery usage; thus the discovery time interval should be planned carefully. The application of BLE communication for RSSI-based vehicle detection and classification has not been considered previously by other authors. This study involves detailed verification of the above-mentioned solution in real-traffic conditions.

Another important drawback of the existing methods lies in limited accuracy of vehicle detection and classification. To overcome this drawback, a new ensemble of classifiers was designed in this study, which accurately detects vehicles and recognizes three vehicle classes based on RSSI data collected from multiple smartphones.

The existing methods utilize single classifiers to detect vehicle and recognize its class. In the related works, the RSSI-based vehicle classification was implemented with use of various classification methods: artificial neural networks [22], k-Nearest Neighbor (k-NN), support vector machine (SVM) [23], decision trees [31], and logistic regression [32]. A SVM method was adopted in [23] to train vehicle classification models and categorize vehicles into two classes (passenger car and truck). The state-of-the-art algorithms are trained using raw data [23] or a set of predefined features [31, 32]. To the best authors' knowledge, classifier ensembles have not been

previously adapted to deal with the RSSI-based road traffic monitoring tasks.

In machine learning literature, various ensemble methods are presented, which combine several classifier systems that use different models or datasets [33]. Several bootstrapping methods were considered (bagging or boosting), which allows us to optimize classifier ensembles [34] or merge classifier decision [35]. Research in [36, 37] shows that combined classifier can outperform the best individual classifier under some conditions (e.g., majority voting by a group of independent classifiers). That works have motivated the approach described in this paper, which involves design and verification of classifier ensembles for traffic monitoring with use of the RSSI data. In comparison with the state-of-the-art methods that are based on single classifiers, the proposed approach enabled more accurate vehicle detection and classification.

3. Proposed Method

The proposed vehicle detection and classification system utilizes RSSI data collected by mobile devices (e.g., smartphones) in a predetermined region on the side of the road. Mobile devices measure signal strength when receiving radio frames from BLE beacons across the street. The RSSI values together with information about position of the device are transmitted to a server, which performs data aggregation and classification.

Structure of the proposed traffic monitoring system is presented in Figure 1. It should be noted that the introduced system structure, which includes BLE beacons and mobile devices, has not been considered in the literature. The BLE beacons are installed at different heights because such arrangement is suitable for vehicle classification, i.e., recognition of personal cars, semitrucks, and trucks [32]. Beacons use the iBeacon protocol [29] to broadcast frames. The mobile devices on the opposite side of the road use BLE communication to collect incoming beacon frames and evaluate their RSSI. Position of the device can be determined based on both the RSSI information and the GPS signal. The collected data are transmitted to a server via cellular network or WiFi communication.

According to the iBeacon protocol, three fields in the broadcasted frames are available that identify the sending

```

1 while mobile device is active do
2 begin
3   t:= current time
4   repeat
5     if new beacon frame received and RSSI > threshold then
6       add record (time, position, beacon_ID , RSSI) to buffer
7     until (current time - t) > T
8     send records from buffer to server
9 end

```

ALGORITHM 1: Mobile device operations.

```

1 create table Records with columns time, position, beacon_ID , RSSI
2 create table Events with columns time, event_type
3 at each time step do
4   if New_records received then
5     begin
6       Records:= Records union New_records
7       time_min := select min(time) from New_records
8       time_min := time_min - window_size
9       Selected_records := select * from Records where time >= time_min
10      Aggregates:= aggregation(Selected_records , window_size)
11      New_events := events_recognition (Aggregates)
12      Events = events_update (Events, New_Events )
13    End

```

ALGORITHM 2: Server operations.

beacon: UUID (universally unique identifier), Major, and Minor value. UUID contains 32 hexadecimal digits, split into 5 groups, separated by hyphens. The iBeacon standard requires also Major and Minor value to be assigned. Those two values help to identify beacons with greater accuracy than using the UUID alone. The Minor and Major values are unsigned integers between 0 and 65535. The purpose of the UUID is to distinguish beacons in a given network from beacons in other networks. For instance, the same UUID can be used for all beacons in a traffic monitoring system, which covers many detection areas. Major values are intended to identify a group of beacons; e.g., all beacons in a certain detection area can be assigned a unique Major value. Finally, Minor values are intended to distinguish an individual beacon. The Minor value can be used for distinguishing individual beacons installed at different heights within a detection area. In this paper, the 3-tuple of UUID, Major, and Minor fields is referred to as beacon ID.

In this study new algorithms (Algorithms 1–5) were designed and implemented to enable accurate vehicle detection and classification with use of BLE beacons and mobile devices. Details of the operations performed by mobile device are presented in Algorithm 1. The received beacon frames are ignored if the RSSI is below a predetermined threshold. In the opposite situation a new data record is created and written to a buffer. The data record contains information about frame reception time, device position, ID of frame sender (beacon), and RSSI value. The content of the buffer is periodically sent

to the server. Frequency of these data transfers is controlled by parameter T . It should be noted that the beacon frames collection and data transfer to server can be performed in parallel if appropriate hardware solution is available.

The objective of server operations (Algorithm 2) is to recognize event type based on the data records delivered from mobile devices. The event type determines if the monitored road section was empty or a car was present in this section during transmission of beacon frames. Additionally, the type of the event indicates class of detected vehicle (personal car, semitruck, or truck). According to the proposed method, the type of the event is recognized using a classifier ensemble (Algorithm 4).

Before execution of the classification procedure, the input data are aggregated. The proposed aggregation procedure is based on so-called sliding window concept [38] (Algorithm 3). It means that if a new data record is received, which contains RSSI value for time t , then the aggregation operation refers to a collection of data records for which the frame reception time t' satisfies condition $t - w \leq t' \leq t$, where w is size of the time window. Such collection of data records is used to calculate aggregates (statistics) of RSSI values, i.e., minimum, maximum, average, and standard deviation. Separate aggregates are determined for each pair of the transmitter (beacon) position and the receiver (mobile device) position. The positions of beacons do not change; thus they are identified by the beacon ID. In contrast, current position of mobile device is assigned to the nearest reference

```

1  Input: Records, window_size
2  Output: Aggregates
3  create table Aggregates
4    with columns time, min_1_1, max_1_1, ..., min_m_n, max_m_n
5  Times:= Select time from Records
6  for each t in Times do
7  begin
8    for refPos := 1..m do
9      for bID := 1..n do
10     begin
11       RSSI_data := Select RSSI from Records
12         where time is between t - window_size and t
13         and distance(position, refPos) <= d_max
14         and beacon_ID = bID
15       min_refPos_bID := min( RSSI_data )
16       max_refPos_bID := max( RSSI_data )
17     end
18     Insert t, min_1_1, max_1_1, ..., min_m_n, max_m_n into Aggregates
19 End

```

ALGORITHM 3: Aggregation function.

```

1  Input: Aggregates
2  Output: New_events
3  create table New_events with columns time, event_type
4  Times:= Select time from Aggregates
5  for each t in Times do
6  begin
7    votes:= empty array
8    for each classifier in ensemble
9    begin
10     [a, b] := classifier_range
11     data:= Select min_a_1, max_a_1, ..., min_b_n, max_b_n
12       from Aggregates where time = t
13     event_type := classifier(data)
14     votes[event_type] := votes[event_type] + classifier_weight
15   end
16   event_type := arg max (votes[event_type])
17   Insert t, event_type into New_events
18 End

```

ALGORITHM 4: Events_recognition function.

```

1  Input: Events, New_events
2  Output: Events
3  New_times := Select time from New_events
4  Times:= Select time from Events
5  for each t in New_times do
6  begin
7    event:= Select * from New_events where time = t
8    if t is in Times then Delete from Events where time = t
9    Insert event into Events
10 End

```

ALGORITHM 5: Events_update function.

position. It should be noted that a set of reference positions in the region of interest, on the side of the road, has to be determined in advance.

Details of the proposed data aggregation procedure are presented by the pseudocode in Algorithm 3. For the sake of simplicity, it was assumed in this pseudocode that only two statistics are to be calculated (maximum and minimum). In practical applications the number of statistics has to be larger, as discussed in Section 4. The symbols min_refPos_bID and max_refPos_bID in Algorithm 3 denote the minimum and maximum RSSI value determined for frames sent from beacon bID and received by a mobile device close to reference position $refPos$ in time window $[t - w, t]$. The statement that a mobile device is close to a reference position means that its distance to the reference position is below d_max . It should be noted that d_max is set to be lower than half of the minimum distance between reference positions; thus each mobile device is assigned to single reference position. The number of reference positions and the number of beacons in Algorithm 3 are denoted by m and n , respectively.

As it was already mentioned above in this section, the type of the event (which relates to vehicle presence and class) is recognized based on the aggregated RSSI data, by using a classifier ensemble (Algorithm 4). The proposed ensemble consists of classifiers that are fed with various subsets of the aggregated data. A different set of the reference positions, for which the RSSI data are collected, is assigned to each classifier in the ensemble. Hereinafter, this set will be referred to as the classifier range. The reference positions are identified by natural numbers $1, \dots, m$. Thus, the classifier range can be defined by a pair $[a, b]$, where $1 \leq a \leq m$, and $a \leq b \leq m$. The range $[a, b]$ means that the input dataset of the corresponding classifier includes the aggregates (e.g., min_refPos_bID and max_refPos_bID) that were determined for the reference positions $refPos = a, \dots, b$. In case of range $[1, m]$ the classifier utilizes the complete dataset. On the other hand, the classifier's input dataset includes the RSSI readings for only one reference position when $a = b$.

For each classifier in the ensemble a weight is determined, which corresponds to number of the classifier's votes. The total number of votes for a given event type is calculated by adding the weights of the classifiers that have recognized this particular event type. As a result, the event type, which receives the highest total number of votes, is selected. In case of a tie the class which has higher a priori probability is selected. Weights of the classifiers are adjusted during training procedure with use of the evolutionary strategy [39].

In this study, application of various machine learning algorithms was considered for implementation of the proposed ensemble (support vector machines, random forest, probabilistic neural network, and k-nearest neighbors' algorithm) [31, 40]. A separate training dataset, which includes classes (i.e., event types) determined by human observer, was used to train the classifiers.

After the events are recognized, an update of the vehicles classification and detection results is conducted in accordance with Algorithm 5. This update is necessary because the new results can be related to time moments for which some events have already been recognized. The new results are

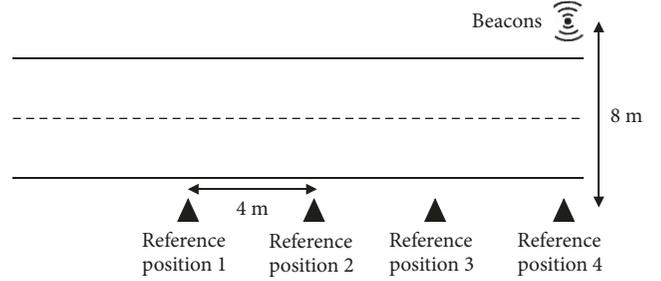


FIGURE 2: Test site with reference positions and beacons.

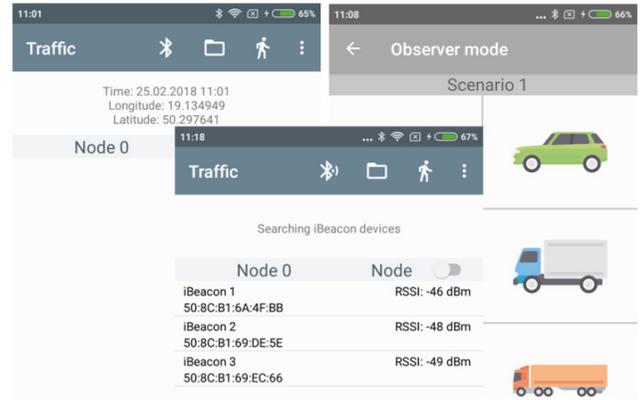


FIGURE 3: Mobile application used for data collection.

more credible as they take into account additional, recently collected data. Thus, the previous results are deleted. Finally, the table *Events* includes the information about event type for all time points covered by the available RSSI dataset. It should be also noted that in this study four event types are considered (empty road, presence of personal car, semitruck, and truck)

4. Experimental Results

Usefulness of the proposed vehicle detection and classification method was verified during experiments in real-world traffic conditions. A schema of the test site, as well as distances between reference positions and beacons, is presented in Figure 2. Three BLE beacons were installed on road side at height of 50, 100, and 200 centimeters above the road surface. This configuration was selected, as providing the most promising results, on the basis of preliminary tests [32]. On the opposite side of the road, four reference points were determined in equal distances of 4 meters. In this area the RSSI measurements were conducted using four smartphones Redmi 3S held at a height of about 1 meter near to the reference positions. The data were collected in a period of two hours. During that period, more than 400 vehicles have passed through the analyzed road section. A mobile application was developed to enable effective collection of the experimental data (Figure 3). Additional mobile devices were used by observers to record the events related to presence of vehicles in front of the reference locations, with recognition of three vehicle classes (personal car, semitruck,

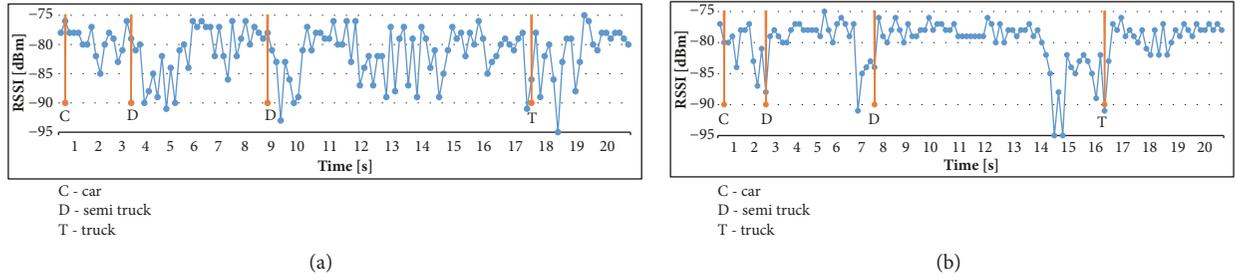


FIGURE 4: Example of collected data: (a) reference position 1 and (b) reference position 4.

and truck). All the mobile devices were synchronized via NTP protocol.

Examples of collected records for two different reference positions are presented in Figure 4. The vertical red lines in Figure 4 show the time instances when passing vehicles were registered by the observers. The labels below vertical lines denote class of the vehicles. These results show that the vehicles cause visible changes of RSSI for both locations. Moreover, the signal noise increases with distance between beacons and mobile device (Figure 4(a)).

For the experimental purposes, the collected data were divided into training and test datasets. The experiments were conducted to evaluate the accuracy of automatic vehicle classification based on the collected data, with use of different machine learning algorithms, i.e., support vector machines (SVM), random forest (RF), probabilistic neural network (PNN), and k -nearest neighbors' algorithm (KNN).

The SVM algorithm [41] performs classification tasks by using hyperplanes defined in a multidimensional space. The hyperplanes that separate training data points with different class labels are constructed at the training phase. SVM employs an iterative training procedure to find the optimal hyperplanes having the largest distance to the nearest training data point of any class. The larger distance results in lower generalization error of the classifier.

In case of RF classifier [42], the training procedure creates a set of decision trees from randomly selected subset of training data. Each tree performs the classification independently and "votes" for the selected class. Finally, the votes from different decision trees are aggregated to decide the class of a test object. At this step, the RF algorithm chooses the class having the majority of votes from particular decision trees.

PNN [43] includes three layers of neurons (input layer, hidden layer, and output layer). The neurons in hidden layer determine similarity between test input vector and the training vectors. To evaluate this similarity each hidden neuron uses a Gaussian function, which is centered on a training vector. The hidden neurons are collected into groups: one group for each of the classes. There is also one neuron in the output layer for each class. The output neuron calculates class probability on the basis of values received from all hidden neurons in a given group. As a result, the posterior probability is evaluated for all considered classes. The final decision of the classifier is the class with maximum probability.

KNN algorithm [44] computes distances between the test data point and all training data points in feature space.

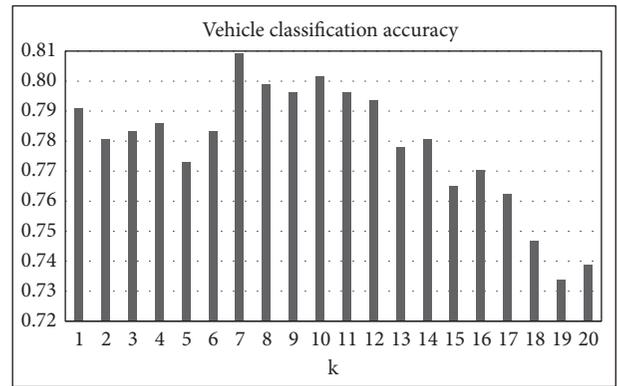


FIGURE 5: Impact of parameter k (number of the nearest neighbors) on accuracy of KNN algorithm.

Afterwards, k training data points with the lowest distances are selected as the nearest neighbors. The test data point is assigned to the class, which is most common among the k -nearest neighbors.

During experiments the classification accuracy was compared for several RSSI-based traffic monitoring approaches, including the proposed solution and the state-of-the-art methods from the literature. This comparison takes into account the method with one receiver [25], solutions with multiple, spatially distributed receivers and single classifier, which detects the vehicles based on a complete RSSI dataset [22, 23], and the new introduced algorithm with the ensemble of classifiers.

Initial experiments were conducted to calibrate parameters of the algorithms. In these experiments, vehicle classification was performed with use of 8 aggregates (minimum, maximum, difference between max. and min., mean, standard deviation, median, Pearson correlation coefficient, and number of received frames). The aggregates were calculated based on the RSSI data collected in four reference positions, in accordance with Algorithm 3.

Accuracy of the KNN algorithm was tested for parameter k (number of the nearest neighbors) in range between 1 and 20. Results of the tests are presented in Figure 5. Based on these results the value $k = 7$, which gave the highest classification accuracy, was selected for further experiments.

Figure 6 shows the classification accuracy that was achieved by using the RF algorithm with different number

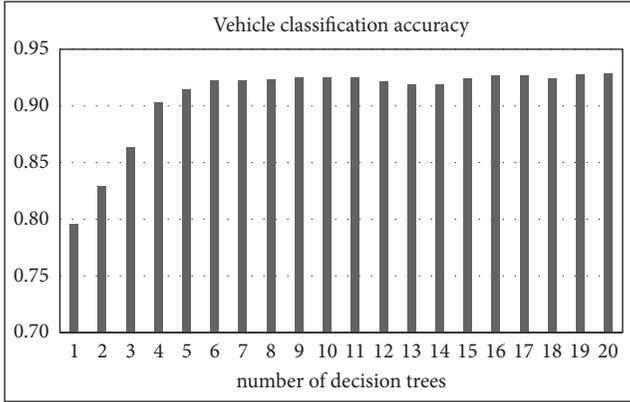


FIGURE 6: Accuracy of random forest algorithm for different number of decision trees.

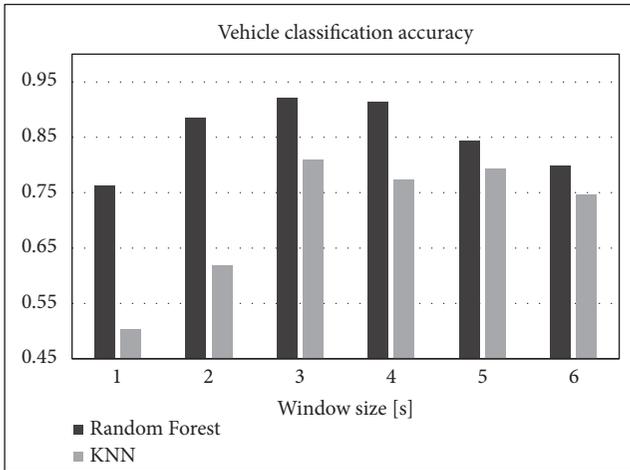


FIGURE 7: Impact of window size parameter on accuracy of RF and KNN algorithms.

of decision trees. It can be observed in these results that the accuracy does not change significantly for the number of decision trees above 5. However, the accuracy achieved for the tree number between 6 and 9 was slightly lower than for the RF with 10 trees. A little decrease of the accuracy was also observed for the tree number above 10. Therefore, during experiments described later in this section the number of decision trees was set to 10. It should be also noted that the complexity of the algorithm increases when using a larger set of the decision trees.

The impact of the window size on vehicle classification accuracy was also examined during the preliminary experiments. The window size was changed from 1 to 6 seconds with steps of 1 second. As shown in Figure 7 for RF and KNN algorithms, the best results were obtained when using the window size of 3 seconds. In case of larger windows the classification accuracy decreases because the data registered for multiple vehicles are aggregated in one window. Similar results were also observed for SVM and PNN algorithms. Thus, the 3-second window was used in further experiments.

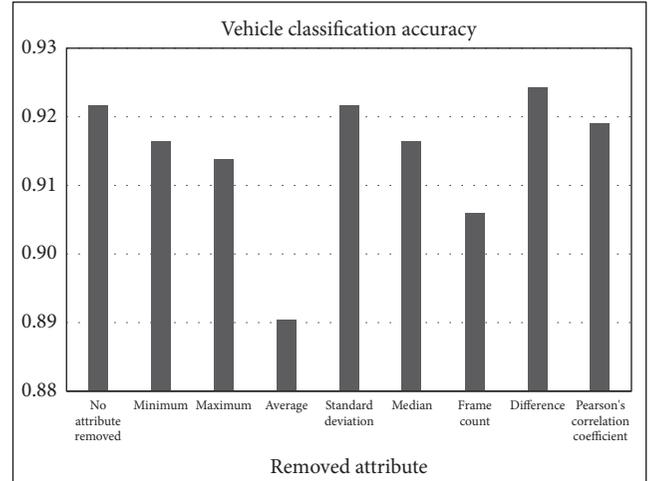


FIGURE 8: Impact of attribute selection on accuracy of RF algorithm.

At the next step, the most effective set of attributes was selected with use of the backward elimination method. Results of the elimination for the RF algorithm are presented in Figure 8. At the beginning, the classification accuracy was tested using full dataset with 8 aggregates. The result of this test is shown by the leftmost bar in Figure 8. Next tests were performed for the 8 datasets that were created by removing particular aggregates (attributes). As shown in Figure 8, an improvement of the vehicle classification accuracy was achieved after deletion of the “difference” attribute (i.e., the difference between maximum and minimum). Thus, the reduced dataset includes 7 aggregates: minimum, maximum, mean, standard deviation, median, Pearson correlation coefficient, and number of received frames. Further elimination did not improve the results. It was verified that the deletion of the “difference” attribute is beneficial for all considered classification algorithms.

Table 1 shows the vehicle detection and classification accuracy obtained for the basic approach, which takes into account the signal strength measured by a single device [25] (in one reference position). These results were obtained after the above-discussed initial search of the best algorithm parameters. As it was already mentioned in previous section, in case of the vehicle classification task four classes of events are considered: empty road, presence of personal car, semitruck, and truck. For the vehicle detection problem two classes are taken into account: empty road and presence of a vehicle. The accuracy (ACC) was calculated as overall accuracy, using the following formula:

$$ACC = \frac{\sum_{i=1}^n C_i}{D} \quad (1)$$

where n is number of classes, C_i is number of items (events) in the test dataset that are correctly assigned to i th class (event type), and D is number of items in test dataset.

It should be also noted that the results in Table 1 are presented for the two classification algorithms that provide the best accuracy. These results firmly show that the most

TABLE 1: Accuracy of vehicle detection and classification based on data collected in one reference position.

Reference position	Vehicle classification accuracy		Vehicle detection accuracy	
	KNN	RF	KNN	RF
1	0.788	0.817	0.8486	0.864
2	0.702	0.699	0.7311	0.801
3	0.725	0.804	0.7807	0.859
4	0.822	0.861	0.8982	0.932

TABLE 2: Accuracy of vehicle classification based on data collected in four reference positions.

Window size [s]	Classification algorithm							
	RF		KNN		PNN		SVM	
	ACC	CK	ACC	CK	ACC	CK	ACC	CK
2	0.885	0.801	0.619	0.287	0.533	0.099	0.525	0.000
3	0.922	0.865	0.809	0.658	0.684	0.403	0.561	0.089
4	0.914	0.853	0.773	0.582	0.802	0.639	0.734	0.533
5	0.843	0.729	0.794	0.630	0.629	0.286	0.538	0.036
6	0.799	0.651	0.747	0.549	0.728	0.512	0.559	0.088

accurate vehicle classification and detection was possible when the mobile device is placed opposite the beacons location (in reference position 4). The results confirm observation that noise in RSSI readings increases with the distance from beacons to mobile device. It should be also noted that the number of RSSI samples that are collected when a vehicle is present between beacons and mobile device decreases with the speed of the vehicle. As a result, lower accuracy is observed for higher speed of vehicles. In the considered test site, the vehicles were slowing down when passing the reference position 1 since this position was close to a crossroad. Thus, the accuracy obtained for reference position 1 is higher than for reference positions 2 and 3.

In further tests, the other approach was considered, which is based on application of multiple receivers and one classifier [22, 23]. According to this approach the vehicles were recognized by single classifier, using the dataset collected in four reference positions. Results of these experiments are shown in Table 2. The classification accuracy (ACC) and Cohen's kappa [45] (CK) is compared in Table 2 for all considered classification algorithms and various sizes of the sliding window. When comparing the results in Table 2 with those in Table 1 it can be observed that the RSSI data collected by multiple devices in several locations along the road enable more accurate vehicle classification. Similar experiments were also conducted for the vehicle detection task and the accuracy of 0.935 was achieved.

The results in Table 2 firmly show that size of the sliding window has a significant impact on the accuracy of vehicle detection and classification. Passing vehicles cause a drop in RSSI level. This drop is longer for trucks and shorter for personal cars. In order to correctly recognize the vehicle, the sliding window has to cover the time when RSSI values are reduced. If the sliding window is too narrow, the lower RSSI values may be registered in entire window for different vehicle classes and thus the classes cannot be correctly recognized. If single classifier is used, a wider window is also helpful

because the drop of RSSI is shifted in time for different reference locations. However, in case of an excessive window size, two successive vehicles can be captured in one window, which results in decreased accuracy of the detection and classification. The best result results were obtained by using the random forest classifier with window size of 3 seconds.

The next step of the research was aimed at increasing the accuracy of vehicle detection by using the proposed classifier ensemble in combination with majority voting, as described in Section 3. It should be noted that the proposed method was used with time step of 1 second and $d_{max} = 1$ meter. During the tests of the ensemble, different ranges of individual member classifiers were taken into account (see Table 3). The input data of individual classifiers were obtained not only from particular reference positions (e.g., Classifier 1 in Ensemble no. 1), but also from a connection of the neighboring positions (e.g., Classifier 1 in Ensemble no. 3). When analyzing the results presented in Table 3 it can be observed that the highest accuracy was achieved for the ensembles of the random forest classifiers. The best ensemble (no. 5) combines the classifiers that are fed with data from two neighboring reference positions (Classifiers 1-3) with the classifier created for reference position 4 (Classifier 4) and the classifier, which utilizes the entire dataset (Classifier 5). Classifier 4 with range [4, 4] was included in the ensemble as it provides the best accuracy when using data from single reference position. The high accuracy was also obtained for Ensembles no. 2 and 6. Results of these ensembles are only slightly worse than those for Ensemble no. 5. This fact shows that the proposed approach achieves high vehicle classification and detection accuracy by combining local classifiers (that utilize data from two neighboring reference positions or single reference position) with the global classifier (which makes decisions based on data collected in all reference positions).

It was noted that the random forest algorithm was about 8.5% more effective than KNN. The proposed method

TABLE 3: Accuracy of vehicle detection and classification with use of the proposed classifier ensemble.

Ensemble no.	Classifier range					Vehicle classification accuracy		Vehicle detection accuracy	
	Clas. 1	Clas. 2	Clas. 3	Clas. 4	Clas. 5	KNN	RF	KNN	RF
1	[1, 1]	[2, 2]	[3, 3]	[4, 4]	-	0.862	0.890	0.906	0.956
2	[1, 1]	[2, 2]	[3, 3]	[4, 4]	[1, 4]	0.862	0.935	0.898	0.961
3	[1, 2]	[2, 3]	[3, 4]	-	-	0.799	0.922	0.854	0.963
4	[1, 2]	[2, 3]	[3, 4]	[4, 4]	-	0.833	0.922	0.898	0.969
5	[1, 2]	[2, 3]	[3, 4]	[4, 4]	[1, 4]	0.846	0.943	0.898	0.977
6	[1, 2]	[2, 3]	[3, 4]	-	[1, 4]	0.825	0.940	0.854	0.969
7	[1, 3]	[2, 4]	-	-	-	0.781	0.911	0.752	0.937
8	[1, 3]	[2, 4]	[4, 4]	-	-	0.836	0.922	0.898	0.958
9	[1, 3]	[2, 4]	[4, 4]	[1, 4]	-	0.846	0.932	0.898	0.966
10	[4, 4]	[1, 4]	-	-	-	0.846	0.924	0.828	0.935

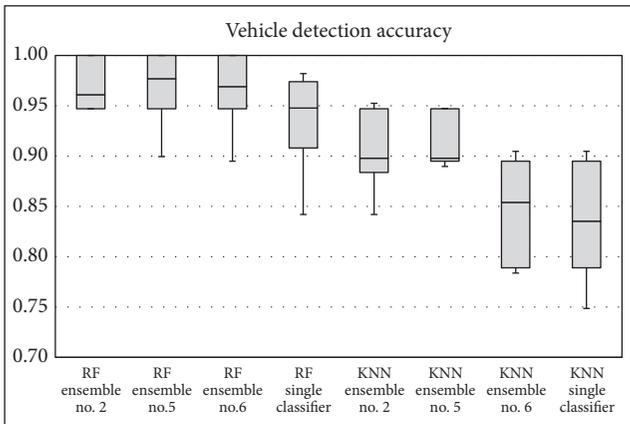


FIGURE 9: Comparison of vehicle detection accuracy for classifier ensembles and for single classifiers.

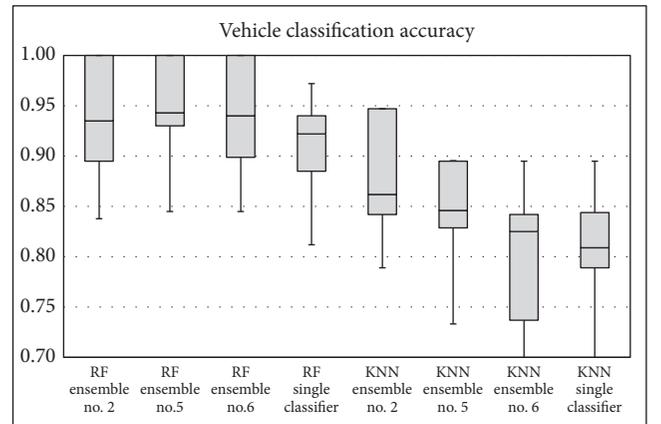


FIGURE 10: Comparison of vehicle classification accuracy for classifier ensembles and for single classifiers.

achieves the accuracy above 97% for vehicle detection task and above 94% in case of the vehicle classification task. It means that the introduced classifier ensemble provides better results than the state-of-the-art methods that utilize individual classifiers (see Tables 1 and 2).

Results obtained for the best classifier ensembles and for the individual (single) classifiers are compared in Figures 9 and 10. The box plots show minimum, first quartile, median, third quartile, and maximum of the accuracy values for 30 tests. For each test different training and testing datasets were selected from the measurement data. In these results significant differences of the accuracy are visible when comparing the single classifiers with their ensemble counterparts. Similarly, the accuracy differences are significant when comparing the RF classifiers with KNN classifiers. It should be also noted that the accuracies achieved by the best RF ensembles do not differ significantly. Thus, selection among these ensembles should be considered as a tuning of the proposed method.

The higher accuracy of RF ensemble can be explained by the fact that the RF algorithm has several features, which enable effective training of the classifier. According to this algorithm, all decision trees in the forest are created by

using randomly selected subsets of the training dataset. The random selection applies to both the events (rows) and the aggregates (columns). Each decision tree further divides the training data into smaller subsets until the subsets are small or all events in these subsets belong to one class. In contrast to RF, the other compared algorithms (including KNN) perform the training procedures with use of the complete training dataset.

5. Conclusions

The proposed vehicle detection and classification approach uses mobile devices (smartphones) and Bluetooth beacons for road traffic monitoring. It allows detecting three classes of vehicles by analyzing strength of radio signal received from BLE beacons that are installed at different heights by the road. This approach is suitable for crowd sourcing applications aimed at reducing travel time, congestion, and emissions. Advantages of the introduced method were demonstrated during experimental evaluation in real-traffic conditions. Extensive experiments were conducted to test different classification approaches and data aggregation methods. In comparison with state-of-the-art RSSI-based vehicle detection

methods, higher accuracy was achieved by introducing a dedicated ensemble of random forest classifiers with majority voting.

The presented solution can be extended to several beacons installed along the road to obtain information concerning vehicle velocity and direction. Another interesting topic is related to data preprocessing on mobile devices in order to reduce the communication effort. Finally, additional studies will be necessary to introduce methods that can be used to activate the Bluetooth modules and beacons when it is necessary and reduce the energy consumption.

Data Availability

The data used to support the findings of this study are included within the supplementary information file.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The research was supported by the National Centre for Research and Development (NCBR) [Grant no. LIDER/18/0064/L-7/15/NCBR/2016].

Supplementary Materials

The supplementary material file (csv) includes a raw RSSI dataset where “id” denotes number of measurement, “Node” is an identifier of mobile device (receiver), “iBeacon” is an identifier of beacon (transmitter), “RSSI” is the measured RSSI value, “Class” describes type of observed event (E: empty road, C: personal car, D: semitruck, and T: truck), and “Flag” indicates the measurements for which the events were recorded (symbol “+”). (*Supplementary Materials*)

References

- [1] H. Chang, Y. Wang, and P. A. Ioannou, “The use of microscopic traffic simulation model for traffic control systems,” in *Proceedings of the 2007 International Symposium on Information Technology Convergence, ISITC 2007*, pp. 120–124, November 2007.
- [2] M. Bernas, B. Płaczek, P. Porwik, and T. Pamuła, “Segmentation of vehicle detector data for improved k-nearest neighbours-based traffic flow prediction,” *IET Intelligent Transport Systems*, vol. 9, no. 3, pp. 264–274, 2014.
- [3] I. Ahmad, R. M. Noor, I. Ali, M. Imran, and A. Vasilakos, “Characterizing the role of vehicular cloud computing in road traffic management,” *International Journal of Distributed Sensor Networks*, vol. 13, no. 5, 2017.
- [4] B. Płaczek, “A self-organizing system for urban traffic control based on predictive interval microscopic model,” *Engineering Applications of Artificial Intelligence*, vol. 34, pp. 75–84, 2014.
- [5] M. Karpiński, A. Senart, and V. Cahill, “Sensor networks for smart roads,” in *Proceedings of the 4th Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom '06)*, pp. 310–314, IEEE, Pisa, Italy, March 2006.
- [6] G. Chatzimilioudis, A. Konstantinidis, C. Laoudias, and D. Zeinalipour-Yazti, “Crowdsourcing with smartphones,” *IEEE Internet Computing*, vol. 16, no. 5, pp. 36–44, 2012.
- [7] R. Prabha and M. G. Kabadi, “KNODET: A Framework to Mine GPS Data for Intelligent Transportation Systems at Traffic Signals,” in *Proceedings of the 2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT)*, pp. 85–89, Bangalore, India, March 2017.
- [8] Y. Ma, L. Zhou, Z. Gu, Y. Song, and B. Wang, “Channel Access and Power Control for Mobile Crowdsourcing in Device-to-Device Underlaid Cellular Networks,” *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 7192840, 13 pages, 2018.
- [9] X. Zhang, Z. Yang, W. Sun et al., “Incentives for mobile crowd sensing: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 54–67, 2016.
- [10] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, “A survey of mobile phone sensing,” *IEEE Communications Magazine*, vol. 48, no. 9, pp. 140–150, 2010.
- [11] W. Z. Khan, Y. Xiang, M. Y. Aalsalem, and Q. Arshad, “Mobile phone sensing systems: a survey,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 402–427, 2013.
- [12] R. K. Ganti, F. Ye, and H. Lei, “Mobile crowdsensing: current state and future challenges,” *IEEE Communications Magazine*, vol. 49, no. 11, pp. 32–39, 2011.
- [13] A. T. Campbell, S. B. Eisenman, N. D. Lane et al., “The rise of people-centric sensing,” *IEEE Internet Computing*, vol. 12, no. 4, pp. 12–21, 2008.
- [14] N. Maisonneuve, M. Stevens, M. E. Niessen, and L. Steels, “NoiseTube: Measuring and mapping noise pollution with mobile phones,” *Information Technologies in Environmental Engineering*, pp. 215–228, 2009.
- [15] C. Costa, C. Laoudias, D. Zeinalipour-Yazti, and D. Gunopulos, “SmartTrace: Finding similar trajectories in smartphone networks without disclosing the traces,” in *Proceedings of the 2011 IEEE 27th International Conference on Data Engineering, ICDE 2011*, pp. 1288–1291, April 2011.
- [16] J. Gomez, J. C. Torrado, and G. Montoro, “Using Smartphones to Assist People with Down Syndrome in Their Labour Training and Integration: A Case Study,” *Wireless Communications and Mobile Computing*, vol. 2017, Article ID 5062371, 15 pages, 2017.
- [17] C. Wu, Z. Yang, and Y. Liu, “Smartphones based crowdsourcing for indoor localization,” *IEEE Transactions on Mobile Computing*, vol. 14, no. 2, pp. 444–457, 2015.
- [18] S. Matyas, C. Matyas, C. Schlieder, P. Kiefer, H. Mitarai, and M. Kamata, “Designing location-based mobile games with a purpose: Collecting geospatial data with cityexplorer,” in *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology, ACE 2008*, pp. 244–247, December 2008.
- [19] H. Aly, A. Basalamah, and M. Youssef, “Robust and ubiquitous smartphone-based lane detection,” *Pervasive and Mobile Computing*, vol. 26, pp. 35–56, 2016.
- [20] E. Koukoumidis, L.-S. Peh, and M. R. Martonosi, “SignalGuru: leveraging mobile phones for collaborative traffic signal schedule advisory,” in *Proceedings of the 9th International Conference*

- on *Mobile Systems, Applications, and Services*, pp. 127–140, July 2011.
- [21] A. Thiagarajan, L. Ravindranath, K. LaCurts et al., “VTrack: accurate, energy-aware road traffic delay estimation using mobile phones,” in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (SenSys '09)*, pp. 85–98, November 2009.
- [22] M. Won, S. Zhang, and S. H. Son, “WiTraffic: Low-cost and non-intrusive traffic monitoring system using WiFi,” in *Proceedings of the 26th International Conference on Computer Communications and Networks, ICCCN 2017*, pp. 1–9, IEEE, August 2017.
- [23] M. Haferkamp, M. Al-Askary, D. Dorn et al., “Radio-based Traffic Flow Detection and Vehicle Classification for Future Smart Cities,” in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, Sydney, NSW, Australia, 2017.
- [24] G. Horvat, D. Šoštarić, and D. Žagar, “Using radio irregularity for vehicle detection in adaptive roadway lighting,” in *Proceedings of the 35th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2012*, pp. 748–753, IEEE, May 2012.
- [25] S. Roy, R. Sen, S. Kulkarni, P. Kulkarni, B. Raman, and L. K. Singh, “Wireless across road: RF based road traffic congestion detection,” in *Proceedings of the 2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011)*, pp. 1–6, IEEE, January 2011.
- [26] N. Kassem, A. E. Kosba, and M. Youssef, “RF-based vehicle detection and speed estimation,” in *2012 IEEE 75th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, IEEE.
- [27] X. Li and J. Wu, “A new method and verification of vehicles detection based on RSSI variation,” in *2016 10th International Conference on Sensing Technology (ICST)*, pp. 1–6, IEEE.
- [28] P. Mestre, R. Guedes, P. Couto, J. Matias, J. C. Fernandes, and C. Serodio, “Vehicle Detection for Outdoor Car Parks using IEEE802.15.4,” *Lecture Notes in Engineering and Computer Science*. Newswood Limited – IAENG, 2013.
- [29] Apple Inc.. Getting Started with iBeacon. Tech. Rep. 1.0. June 2014.
- [30] A. Lindemann, B. Schnor, J. Sohre, and P. Vogel, “Indoor positioning: A comparison of WiFi and Bluetooth Low Energy for region monitoring,” in *Proceedings of the International Joint Conference on Biomedical Engineering Systems and Technologies, Volume 5: HEALTHINF*, pp. 314–321, Rome, Italy, February 2016.
- [31] V. Martsenyuk, K. Warwas, K. Augustynek et al., “On multivariate method of qualitative analysis of Hodgkin-Huxley model with decision tree induction,” in *Proceedings of the 2016 16th International Conference on Control, Automation and Systems (ICCAS)*, pp. 489–494, Gyeongju, South Korea, October 2016.
- [32] M. Bernas, B. Płaczek, and W. Korski, “Wireless Network with Bluetooth Low Energy Beacons for Vehicle Detection and Classification,” in *CN 2018: Computer Networks*, P. Gaj, M. Sawicki, G. Suchacka, and A. Kwiecień, Eds., vol. 860 of *Communications in Computer and Information Science*, pp. 429–444, Springer, 2018.
- [33] M. Woźniak, M. Graña, and E. Corchado, “A survey of multiple classifier systems as hybrid systems,” *Information Fusion*, vol. 16, no. 1, pp. 3–17, 2014.
- [34] G. Marcialis and F. Roli, “Fusion of face recognition algorithms for video-based surveillance systems,” in *Multisensor Surveillance Systems: The Fusion Perspective*, G. L. Foresti, C. Regazzoni, and P. Varshney, Eds., pp. 235–250, 2003.
- [35] R. Polikar, “Ensemble learning,” *Scholarpedia*, vol. 3, no. 12, article 2776, 2008.
- [36] G. Brown, J. Wyatt, R. Harris, and X. Yao, “Diversity creation methods: a survey and categorisation,” *Information Fusion*, vol. 6, no. 1, pp. 5–20, 2005.
- [37] M. Bernas and B. Płaczek, “Fully connected neural networks ensemble with signal strength clustering for indoor localization in wireless sensor networks,” *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 403242, 2015.
- [38] M. Lewandowski, T. Orczyk, and B. Płaczek, “Human activity detection based on the iBeacon technology,” *Journal of Medical Informatics Technologies*, vol. 25, 2016.
- [39] H.-G. Beyer and H.-P. Schwefel, “Evolution strategies—A comprehensive introduction,” *Natural Computing*, vol. 1, no. 1, pp. 3–52, 2002.
- [40] M. R. Berthold, N. Cebon, F. Dill et al., “KNIME: The Konstanz Information Miner,” in *Data Analysis. Machine Learning and Applications. Studies in Classification. Data Analysis. and Knowledge Organization*, C. Preisach, H. Burkhardt, L. Schmidt-Thieme, and R. Decker, Eds., Springer, Berlin, Germany.
- [41] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, “New support vector algorithms,” *Neural Computation*, vol. 12, no. 5, pp. 1207–1245, 2000.
- [42] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [43] D. F. Specht, “Probabilistic neural networks,” *Neural Networks*, vol. 3, no. 1, pp. 109–118, 1990.
- [44] D. W. Aha, D. Kibler, and M. K. Albert, “Instance-Based Learning Algorithms,” *Machine Learning*, vol. 6, no. 1, pp. 37–66, 1991.
- [45] N. C. Smeeton, “Early History of the Kappa Statistic,” *Biometrics*, vol. 41, no. 3, article 795, 1985.



Hindawi

Submit your manuscripts at
www.hindawi.com

