Computer Networks 217 (2022) 109326

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Deep learning approaches for vehicle type classification with 3-D magnetic sensor

Burak Kolukisa [a,*], Veli Can Yildirim [b], Bahadir Elmas [c], Cem Ayyildiz [b], Vehbi Cagri Gungor [a]

[a] Department of Computer Engineering, Abdullah Gül University, Kayseri, Turkey
[b] Department of R&D, GOHM Electronics, Mugla, Turkey
[c] Department of Statistics, Mimar Sinan Fine Arts University, Istanbul, Turkey

ARTICLE INFO

Keywords:
Intelligent Transportation Systems
Magnetic sensor
Vehicle classification
Deep learning

ABSTRACT

In the Intelligent Transportation Systems, it is crucial to determine the type of vehicles to improve traffic management, increase human comfort, and enable future development of transport infrastructures. This paper presents a deep learning-based vehicle type classification approach for intermediate road traffic. Specifically, a low-cost, easy-to-install, battery-operated 3-D traffic sensor is designed and developed. In addition, a total of 376 vehicle samples are collected, and the vehicles are identified into three different classes according to their structures: light, medium, and heavy. Firstly, an oversampling method is applied to increase the number of samples in the training set. Then, the signals are converted into time series for LSTM and GRU and 2-D images for transfer learning models. Finally, soft voting is proposed using the LSTM, GRU, and VGG16, which is the best transfer learning method for vehicle type classification. The developed system is portable, power-limited, battery-operated, and reliable. Comparative performance results show that the soft voting ensemble method using a deep learning classifier improves the accuracy and f-measure performances by 92.92% and 93.42%, respectively. Additionally, the battery lifetime of the developed magnetic sensor node can work for up to 2 years.

1. Introduction

In recent years, rapid population growth and vehicle demand have significantly increased the number of vehicles in traffic, particularly in metropolitan cities. This situation is increasing the need for Intelligent Transportation Systems (ITS). Vehicle detection and classification have become an indispensable part of ITS to increase human comfort, improve traffic management, and enable future development of transport infrastructures. Significant investments are being made and used in the development, implementation, and maintenance of traffic monitoring systems in many countries [1]. Recently, in spite of the technical challenges, several vehicle type classification systems have been developed based on accelerometers [2], acoustic sensors [3], camera [4], hybrid methods [5], LIDAR [6], loop detectors [7], piezoelectric sensors [8], vibration sensors [9], and magnetic sensors [10–19]. These technologies have important specific characteristics and requirements, such as sensor types, hardware settings, setup processes, parameter settings, operating environments, weather/noise resistance, battery lifetime, and even maintenance and installation costs.

In the literature, it is shown that magnetic sensors are preferred for vehicle classification due to their advantages, such as strong climate adaptation, small size, easy installation, and low cost. In Table 1, the most recent studies that use magnetic sensor technologies and classify vehicle types are compared and summarized. Specifically, the study [15] focused on the classification of similar vehicle sizes using multiple sensor nodes and utilizes an XGBoost method. The study [16] proposed K-Nearest Neighbor (K-NN) method for classification of vehicle types. In [17] the magnetic waveforms obtained from two sensor nodes were fussioned, and a Support Vector Machine (SVM) was used for classification. The study [19], focused on classifying vehicles with the convolutional neural network (CNN) method. In contrast to the existing studies, in this paper, deep learning approaches are proposed for vehicle type classification using a single 3-D magnetic sensor node. The main contributions of this study are listed below:

- To the best of our knowledge, this article is the first study focusing on deep learning methods and soft voting ensemble techniques to classify vehicle types with a single 3-D magnetic sensor node.

* Corresponding author.
E-mail addresses: burak.kolukisa@agu.edu.tr (B. Kolukisa), veli.yildirim@gohm.com.tr (V.C. Yildirim), bahadir.elmas@msgsu.edu.tr (B. Elmas), ca@gohm.com.tr (C. Ayyildiz), cagri.gungor@agu.edu.tr (V. Cagri Gungor).

https://doi.org/10.1016/j.comnet.2022.109326
Received 24 September 2021; Received in revised form 2 July 2022; Accepted 24 August 2022
Available online 28 August 2022

Table 1
Summary of studies for vehicle type classification with magnetic sensor technologies.

Paper	Sample size	# of VT	Methodology	Cost	Energy efficiency	Single node	Deep learning	Ensemble classifier	Year
[10]	5.837	3	Length Threshold	×	×	×	×	×	2013
[11]	188	4	SVM	$50	×	×	×	×	2014
[12]	253	5	Hierarchical Three	×	×	✓	×	×	2015
[13]	20.353	4	Fusion Algorithm	×	×	×	×	×	2017
[14]	12.085	5	Decision Tree	$80	×	×	×	×	2017
[15]	1.442	4	XGBoost	×	×	✓	×	×	2018
[16]	300	6	K-NNs	×	×	✓	×	×	2018
[17]	732	3	SVM	×	×	✓	×	×	2019
[18]	412	5	SVM	×	×	✓	×	×	2019
[19]	6.042	7	CNN	×	×	✓	✓	×	2020
PM	376	3	LSTM + GRU + VGG16	$25	✓	✓	✓	✓	2021

VT: Vehicle Type, PM: Proposed Model.

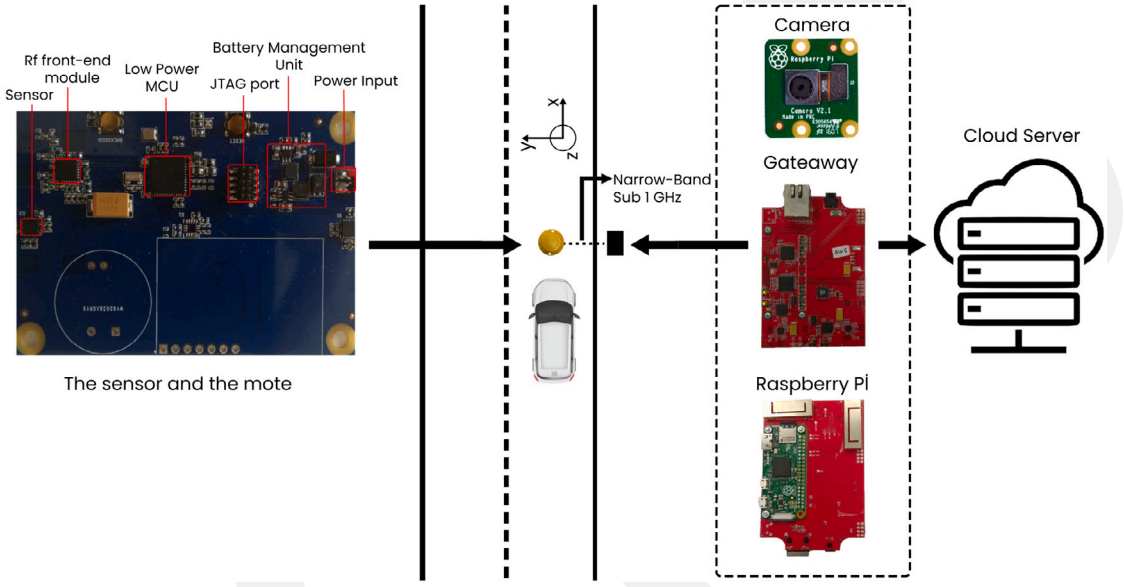


Fig. 1. The illustration diagram of the proposed system using the 3-D magnetic sensor for vehicle type classification.

- The proposed 3-D magnetic sensor node is low-cost ($25), easy-to-install, portable, and battery-operated. It can work for up to 2 years without requiring battery replacement.
- Comparative performance results show that the proposed soft voting ensemble method using a deep learning classifier improves the accuracy and f-measure performances by 92.92% and 93.42%, respectively.

The rest of the paper is organized as follows. In Section 2, the 3-D magnetic sensor node details are introduced. In Section 3, the dataset is introduced as well as the preparation steps are explained. Section 4 elaborately describes the deep learning methodologies and the proposed approach. Then, Section 5 shows the performance results of the models comparatively and discusses the outcomes. Finally, the last section concludes the paper.

2. Magnetic sensor

The proposed system consists of a 3-D magnetic sensor to measure the intensity of magnetic fields, a mote responsible for reading the sensor outputs, a gateway responsible for transmitting the data provided by the sensor mote to the data center, and a web server responsible for analyzing and displaying the data that has been collected. The system is protected by a mechanical capsule from environmental factors. The proposed system diagram has shown in Fig. 1.

2.1. Sensor

In order to extract new features for the classification algorithms, a single 3-D magnetic sensor is used as a sensing unit. The sensor node's features are as follows: The supply voltage is between 0.9 and 3.6 V; the maximum data rate is 5 kbps; the TX power is +30 dBm; the RF frequency is between 863 and 876 MHz; the RF communication distance is between 100 and 700 m; the operation temperature is between −25 °C and 80 °C; the mechanical robustness is up to 10,000 kg; the waterproof IP rating is IP 67; and the battery lifetime is 2 years. The cost analysis of mote is as follows: The MCU (CC1312) costs $7, the RF Module (SE2435L) costs $3, the Magnetic Sensor Array costs $5, the Flash Memory costs $1, the Antenna costs $2, and the PCB costs $7. The proposed sensor node has a total cost of 25$, which is not mentioned in most of the studies. A limited number of recent studies [11,14] mentioned that their sensor costs are 50$ and 80$, respectively, while our sensor node costs much lower compared to these studies.

2.2. Mote

The developed mote is equipped with a CC1312 wireless MCU that has a 48-MHz Cortex-M4F microcontroller, a specific radio controller that is based on Cortex-M0, an ultralow-power 8-bit sensor controller IC, 80 kB of SRAM, UART, I2C, and SPI [20]. The CC1312 wireless MCU, which operates on the 868 MHz band and utilizes the 802.15.4 communication protocol and is responsible for ensuring connectivity

with the gateway. In order to improve the RF power output, the RF front-end module has also been implemented [21]. The built-in software on the mote is used to calibrate the sensor, get measurements from the sensor in real time, process and temporarily store the gathered data, and finally transmit the stored data to the gateway.

2.3. Gateway

The gateway is equipped with the same mote hardware connected to a Raspberry Pi [22]. The data acquired from sensor motes is sent to the data center via the Linux operating system, which is run on the Raspberry Pi. The same type of transceiver is used for the motes connection. The gateway features are as follows: The voltage supply is 5 V; the TX power is +30 dBm; the RF frequency is between 863 and 876 MHz; the RF communication distance is between 100 and 700 m; the operation temperature is between −25 °C and 80 °C; and the maximum number of connected nodes is up to 100.

2.4. Backend

The sensor values are stored along with a reference camera system to facilitate data tagging and store the data for later offline processing. A backend cloud system has been developed. By synchronizing the camera with the magnetic sensor node, the user is able to select the time frame for cropping and tagging the videos with the magnetic sensor node's data. The video dataset is saved in the NoSQL database for future usage and long-term retention.

2.5. Magnetic sensor measurements

Metal materials in the vehicle cause magnetic disturbances, these distortions cause signal changes, and they differ for each vehicle. These differences are mainly due to the structure of the vehicle, and environmental factors. Therefore, with the development of information technologies, it has become possible to predict vehicle types by analyzing magnetic sensor measurements via machine learning approaches. In this study, magnetic signatures of vehicles are obtained using two different measurement methods, i.e. time-dependent measurement method and vector magnitude-dependent measurement method. The first method (time-dependent measurement method) has the advantage of reducing power consumption, since data is received at a certain time interval; there is no need to add a timestamp to the packets sent by narrow band. Therefore, the sensor consumes less power. Nevertheless, it is possible for magnetic field sensors to fail to detect magnetic field changes, which would raise the magnetic signature's error rate. The vector magnitude dependent measurement is used in the second method to take samples based on certain magnetic field changes (for example, every 10 microteslas). The main benefit of this method is the ability to capture all magnetic field changes, and it allows for processing larger sample amounts, which can obtain more accurate signatures. Due to the timestamp information, the amount of data transmitted in a narrow band increases. As a result, the battery consumption also increases.

As for the implementation of the project, it was decided to employ the second approach to detect magnetic changes during a vehicle's entry and exit from the sensor. The first approach was also used to sample the vehicles' movements on the sensor by time-based. This technique helps us to reduce power consumption.

As the vehicle passes over a 3-D magnetic sensor node, magnetic sensor interrupt and record the vehicles' measurements. Then, measurements have been transmitted to the gateway, and sleeps. The sampling frequency of the sensor is set to 400 Hz, which is its maximum frequency. We measured the magnetic distortions of 50 different vehicles using a sensor node mounted on a single-lane road. The threshold (T), as well as the vehicle's reference magnetic distortions X_r, Y_r, and Z_r, are determined, as shown in following equation:

$$\sqrt[2]{(X_o - X_r)^2 + (Y_o - Y_r)^2 + (Z_o - Z_r)^2} > T \quad (1)$$

Based on the magnetic distortions of 50 different vehicles, T is 110. A new vehicle's magnetic distortions are shown as X_o, Y_o, and Z_o while when the result of Eq. (1) for a new vehicle is greater than T, the magnetic distortions of the new vehicle are recorded.

The sensor consumes 20 uA during the sleep period. The average power consumption for a vehicle includes the sensor measurement period and packet transmission, whose durations are 8 and 12 ms. The packet transmission includes pre-processing (10 mA per ms, 2 ms), RX (1 A per ms, 5 ms), TX (20 mA per ms, 3 ms), and post-processing (10 mA per ms, 2 ms). The sensor measurement period uses 40 mA per ms, and packet transmission uses 425 mA per ms. In total, the average power consumption for a vehicle is 5420 mA with a duration of 20 ms.

3. Methods

3.1. Dataset

In this study, the 3-D magnetic sensor node that we developed is mounted on a single-lane road, and data collection is carried out with the help of a camera. While the 3-D magnetic sensor records magnetic disturbances, the camera records images of the vehicles, and these processes occur simultaneously. In total, 376 vehicle samples are collected and labeled. A recorded video is used to label the magnetic disturbances of vehicles. The vehicle types are divided into three categories based on the vehicle's structure: light (motorcycles), medium (passenger cars), and heavy (buses). These classes are also determined by the US Federal Highway Administration (FHWA) [23]. As the vehicle passes over the 3-D magnetic sensor node, it records the signals of the X, Y, and Z axes. The signal lengths of the samples are equalized by padding them with zeros according to the maximum signal length. In total, there are 621 features available for each sample when considering the three axes. Also, the features in the dataset are scaled between 0 and 1 using MinMaxScaler [24].

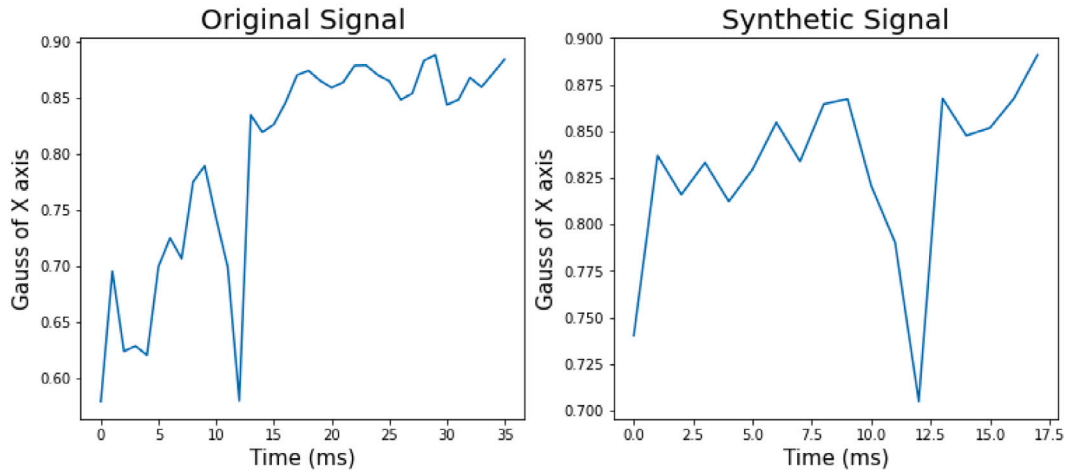
We obtained the test set by selecting 30% of the total set via stratified random sampling and storing the remaining samples as the training set, as shown in Table 2. In addition, for the hyper-parameter optimization, 30% of the training set, which is obtained by the oversampling process, is selected for the validation set via random sampling and storing, and then the remaining samples are selected as a training set.

3.2. Oversampling method (SMOTE)

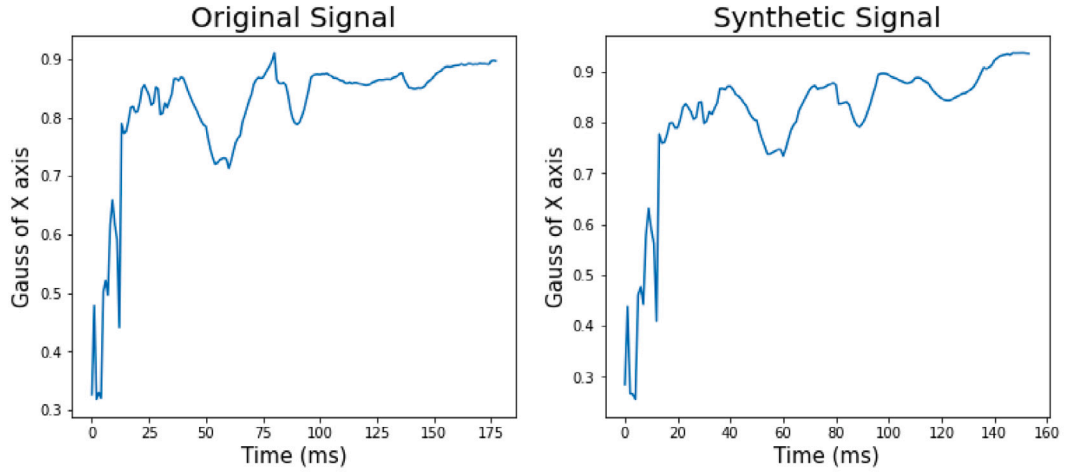
The main reason of applying the oversampling method is that the number of samples is insufficient to train machine learning models, and the unbalanced classes problem causes poor performance. In this study, the Synthetic Minority Oversampling Technique (SMOTE) is applied [25] to increase the number of samples in the training set. The SMOTE algorithm randomly generates new minority samples using the following rules: First, x_i data is randomly selected from the minority classes; then the five data are determined based on the k-nearest neighbors (k-NN) of the selected x_i data; and finally, randomly x_k data is selected. The new synthetic data x_{new} is between the x_i and x_k data according to the λ values, which is randomly selected between (0,1). Formula is shown in equation below:

$$x_{new} = x_i + \lambda(x_k - x_i) \quad (2)$$

The new sample size of the vehicle types is shown in Table 3. In the training dataset, minority classes (light and heavy) are processed by the SMOTE algorithm and generated new synthetic samples as shown in Fig. 2(a) and Fig. 2(b) for light and heavy vehicle, respectively. During the analysis of the new synthetic samples, no absurdity was observed in the wave-forms.



(a) Light vehicle



(b) Heavy vehicle

Fig. 2. The X-axis of the original and synthetic (SMOTE algorithm-processed) signals for (a) light vehicle and (b) heavy vehicle.

Table 2
Sample size of the original vehicle dataset.

Vehicle type	Light	Medium	Heavy
Train	33	207	23
Test	14	89	10
Total	47	296	33

Table 3
Sample size of the new vehicle dataset.

Vehicle type	Light	Medium	Heavy
Train	207	207	207
Test	14	88	10
Total	221	296	217

3.3. Dataset preparation

Transfer learning is a powerful network that tends to learn edges, textures, patterns, and object parts in images [26] and yields satisfactory results. Therefore, in this study, vehicle signals are converted into 2-D images in order to perform vehicle classification by utilizing transfer learning models. Firstly, the lengths of samples are scaled to the maximum signal length of 207. The next step is to color the area under the X, Y, and Z curves. Each color represents a different condition. For example, the pink color represents the area under X curve and the upper area of the Y and Z curves. Table 4 shows the combination of conditions during coloring process. The converted images become 216 × 216 colored images, and the conversion of the signals to the images for the medium vehicle is shown in Fig. 3.

Importantly, the shape of the dataset, i.e., "samples, features", is not suitable for the LSTM and GRU methods. The shape of the dataset is reshaped into "samples, time-steps, and features" which means one sequence is one sample, one sequence has multiple time steps according to the length of the vehicle signal, and one feature corresponds to one variety of signal. The reshaped dataset is "sample size, 207, 3", in which 207 means the vehicle signal length and 3 means the X, Y, and Z-axis.

Table 4
Painting images according to X, Y, and Z Axis-Curves.

No	Condition (Under Axis-Curve)	Color
1	–	Blue
2	X	Purple
3	Y	Red
4	Z	Green
5	X, Y	Black
6	X, Z	Olive
7	Y, Z	Turquoise
8	X, Y, Z	White

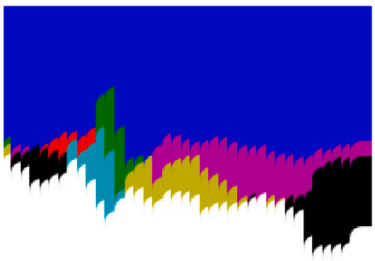


Fig. 3. The X,Y, and Z-Axis of the original vehicle signals converted to image for medium vehicle.

Table 5
Layer configurations of the LSTM/GRU model.

Layer name	CNN models
L1	Masking()
L2	LSTM/GRU (32)
L3	Batch Normalization()
L4	Dropout (0.01)
L5	LSTM/GRU (16)
L6	Batch Normalization()
L7	Dropout (0.01)
L8	LSTM/GRU (8)
L9	Batch Normalization()
L10	Dropout (0.01)
L11	Dense(3)

4. Building classification models

In this study, the test set is obtained by selecting 30% of the total dataset using stratified random sampling, and the remaining samples are used as a training set. The SMOTE method is applied to increase the number of samples in the training set while the size of the test set is set as the same. In addition, for the hyper-parameter optimization, 30% of the new training set is randomly selected for the validation set and the rest is stored as training set. For the LSTM and GRU models, we reshaped the dataset, and for the transfer learning model, we converted the vehicle signals to 2-D images. For this purpose, VGG16, VGG19, Xception, MobilNet, MobilNetV2, DenseNet121, DenseNet169, and DenseNet201 are implemented as deep learning models. The block diagram in Fig. 4 outlines how the classification process in this study is conducted. Models are trained on the training set and validated on the validation set with appropriate hyperparameters and epochs. Lastly, the model is evaluated for the test set. Individual results are obtained for each classifier and three classifiers are implemented for the soft voting ensemble model. All machine learning approaches are implemented with the Python programming language. LSTM, GRU, and transfer learning approaches are implemented using Keras libraries [27], and the SVM method is implemented using scikit-learn libraries [28]. The proposed methods have been presented in the following subsections.

4.1. Support Vector Machine

It is a simple and effective machine learning method often used by the community for classification, regression, and outlier detection. It has a simple method for separating classes, which draws parallel lines between classes. Also, hyper parameter optimization is applied, which ensures the classifiers are trained with the most suitable parameters.

In the implementation of the SVM classifier, hyperparameter optimization is applied using the grid search method. The parameters are determined before the classification method, and the best parameters are obtained using the training and validation sets, and the optimal parameters are used on the test set. The hyper-parameters are optimized for the SVM classifiers as follows: C= {0.1, 1, 10, 100, 1000}, gamma = {1, 0.1, 0.001, 0.0001}, and kernel = {linear, rbf, poly, sigmoid}.

4.2. Recurrent neural networks

A recurrent neural network (RNN) is another variation of a basic neural network that has an internal loop and is good at handling time-sequence data. However, RNN suffers from a vanishing gradient descent problem. LSTM [29], which is a special type of RNN algorithm, has been developed to solve the vanishing gradient descent problem. The main difference is that the LSTM cell has special gates. GRU [30] is quite similar, but it is simpler in structure compared to LSTM. In this study, these two main architectures are used.

In the LSTM and GRU experiments, the input shape of the model is set to “None, 3” and the mask value is set to 0. This is because the samples of different lengths are not suitable for the LSTM and GRU models and must be in the form of a single tensor. For this, the zero-padding process is applied to obtain the same length, and zeros have no meaning for the LSTM/GRU model due to masking. The structure of these LSTM and GRU models on each layer is presented in Table 5. Due to the multi-class classification problem, the activation function of the last layer is softmax. The optimizer is Adam (learning rate = 0.0001) with a categorical cross-entropy loss function. The mini-batch size is selected as 16, the maximum number of epochs is set to 200, and ModelCheckpoint is used to save the model weights in each step. Based on the validation loss, the best epoch is selected for the test set. The parameters of the models have been regularized using batch normalization and dropout.

4.3. Transfer learning

The transfer learning technique is a model that can reuse the information obtained from previous tasks as the starting point for a new task [31]. The fact that the learned qualities are portable is one of the essential advantages that distinguishes deep learning from traditional and shallow learning approaches. It makes deep learning effective in small data problems [32].

In this study, transfer learning methods are used by freezing the convolutional base layers and re-training the top layers (e.g., fully connected MLP layers). The weights of the frozen layers are taken from the models trained on the ImageNet dataset. After the flatten layer of the convolutional base, we applied an MLP network, which has a single hidden layer with 64 neurons. Due to the multi-class classification problem, the activation function of the last layer is softmax. The optimizer is Adam (learning rate = 0.01) with a categorical cross-entropy loss function. The mini-batch size is selected as 16, the maximum number of epochs is set to 50, and ModelCheckpoint is used to save the model weights in each step. Based on the validation loss, the best epoch is selected for the test set. The parameters of the models have been regularized using batch normalization, dropout, and L2-norm regularization. Lastly, due to a limited number of samples available in the training set, data augmentation [33] has been applied. We have increased the variety and number of training images by applying operations, such as rotating, flipping, and cropping.

4.4. Soft voting

Soft voting is an ensemble machine learning model that combines the predicted class labels from multiple models. It predicts the class by averaging each model’s weight equally or weighted specifically by the classifier’s importance. The results obtained from the deep learning models are examined. The VGG16 model shows high performance in classifying light and medium vehicle types and gives very poor results in classifying heavy vehicles. LSTM and GRU models showed success in classifying medium and heavy vehicles and low results for light vehicle types. Considering these situations, the ensemble soft voting methods weights are implemented as follows: for the VGG16 model, the weights

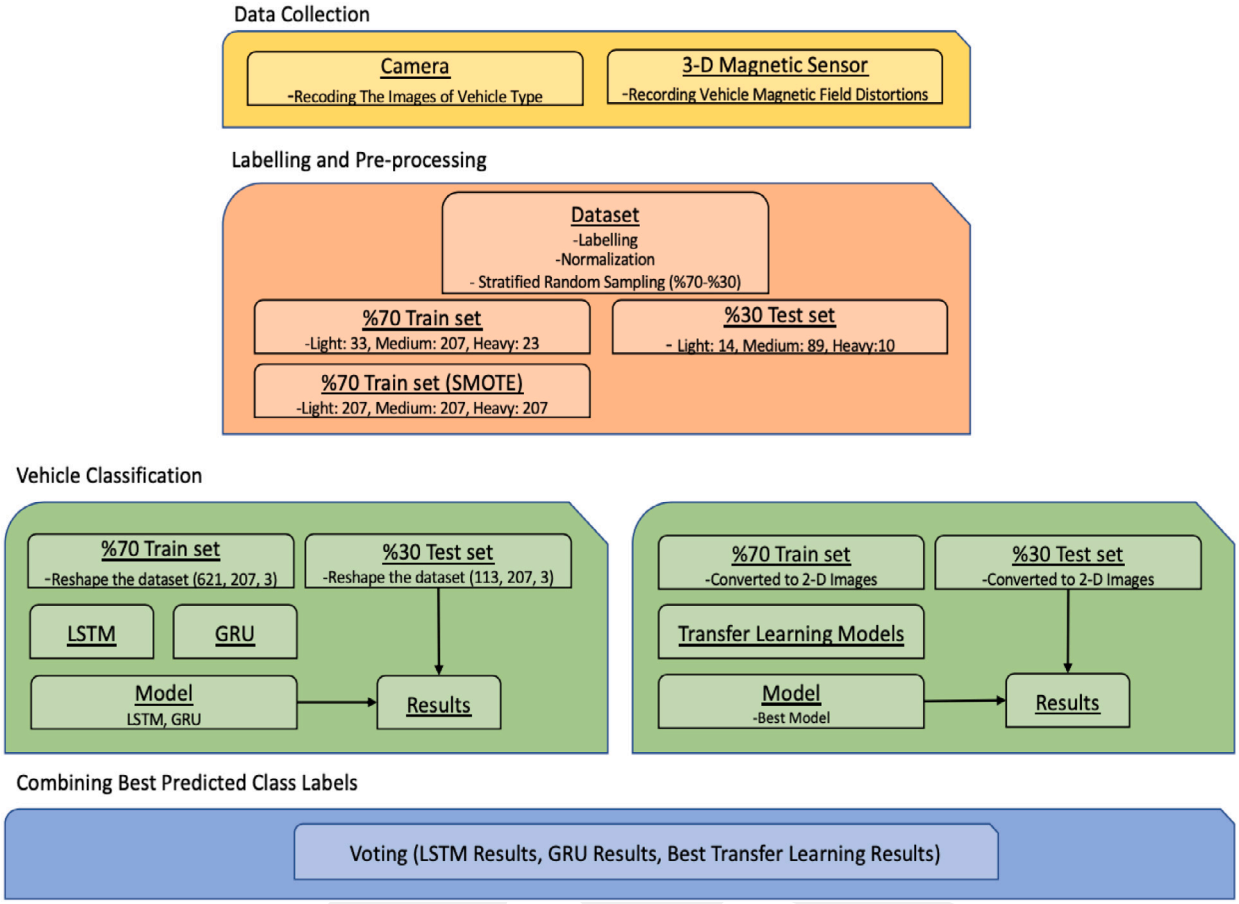


Fig. 4. Block diagram of the classification process.

Table 6
Performance results of classification methods.

Model	Accuracy (%)	F-measure (%)	Precision (%)	Recall (%)
LSTM	74.33	76.63	85.83	84.17
DenseNet201	78.76	76.10	75.40	47.27
DenseNet169	79.64	80.15	81.76	69.62
GRU	81.41	82.29	84.35	76.28
DenseNet121	82.30	71.79	84.33	82.93
MobileNetV2	83.18	82.19	81.55	61.18
SVM	83.18	82.39	82.82	64.99
Xception	84.07	79.33	76.84	49.62
VGG19	88.49	85.18	82.14	63.53
MobileNet	90.26	86.48	83.01	66.29
VGG16	92.03	88.94	92.67	70.00
Ensemble (LSTM + GRU + VGG16)	**92.92**	**93.42**	**94.35**	**91.08**

of the classes light and medium are high, and for the heavy class they are low. For the LSTM and GRU models, the weights of the classes medium and heavy are high, and for the light class they are low (see Table 6).

5. Results & discussion

5.1. Classification methods

In this study, using a camera and a 3-D magnetic sensor node, data was collected on intermediate road traffic by taking 376 vehicle samples and identifying the types of vehicles. LSTM, GRU, SVM, and transfer learning algorithms are applied to the dataset for vehicle type classification. Transfer learning approaches include VGG16, VGG19, Xception, MobilNet, MobilNetV2, DenseNet121, DenseNet169, and DenseNet201. Models are trained on a training set and validated on a validation set, then the model is also tested for the test set. Firstly, the accuracies of LSTM and GRU models are obtained as 74.33% and 81.41%, respectively, with the following optimum hyper-parameters: dropout = 0.2, learning rate = 0.001, L2 norm regularization = 0.0001, and optimizer is Adam. In the next step, the transfer learning approaches are implemented and models are used by freezing the convolutional base layers and re-training the top layers with the following optimum hyper-parameters: dropout = 0.2, learning rate = 0.001, L2 norm regularization = 0.0001, and optimizer set to Adam. The VGG16 model obtained the best results, with an accuracy of 92.03%. Lastly, the SVM model accuracy is obtained as 83.18% with the following optimum hyper-parameters: C = 100, gamma = 0.001, kernel = sigmoid. Table 7 shows all the performance results we obtained in this study for vehicle type classification. While the VGG16 classified light and medium vehicles better, as shown in Fig. 5(a), LSTM and GRU classified medium and heavy vehicles better. Fig. 5(b) and Fig. 5(c) show the performance results of the LSTM and GRU classifiers, respectively.

SVM and some transfer learning methods obtained better accuracy results compared to LSTM and GRU classifiers. Nevertheless LSTM and GRU methods classified heavy class better. For this reason, with the help of an ensemble method, the model results of LSTM, GRU, and VGG16 classifiers results are concatenated and the soft voting ensemble method is applied and an accuracy of 92.92% and f-measure of 93.42% are obtained. The soft voting ensemble technique improves the accuracy score, and also produces a much better f-measure score. Fig. 5(d) shows the performance results of the ensemble soft voting results.

Fig. 6 shows the loss rate of vehicle classification on LSTM, GRU, and VGG16 models. It can be seen in this figure that the best loss rates

		Predicted				
		L	M	H	T	ACC (%)
Actual	Light (L)	14	0	14	14	100
	Medium (M)	0	89	0	89	100
	Heavy (H)	1	8	1	10	10
	Total (T)	15	97	1	113	92.03

(a) Confusion matrix of the VGG16 method.

		Predicted				
		L	M	H	T	ACC (%)
Actual	Light (L)	13	1	0	14	92.85
	Medium (M)	16	62	11	89	69.66
	Heavy (H)	0	1	9	10	90
	Total (T)	29	64	20	113	74.33

(b) Confusion matrix of the LSTM method.

		Predicted				
		L	M	H	T	ACC (%)
Actual	Light (L)	12	2	0	14	85.71
	Medium (M)	9	74	6	89	83.14
	Heavy (H)	0	4	6	10	60
	Total (T)	21	80	12	113	81.41

(c) Confusion matrix of the GRU method.

		Predicted				
		L	M	H	T	ACC (%)
Actual	Light (L)	14	0	0	14	100
	Medium (M)	0	83	6	89	93.25
	Heavy (H)	0	2	8	10	80
	Total (T)	14	85	16	113	92.92

(d) Confusion matrix of the soft voting ensemble method.

Fig. 5. Performance results of VGG16, LSTM, GRU and ensemble methods, respectively.

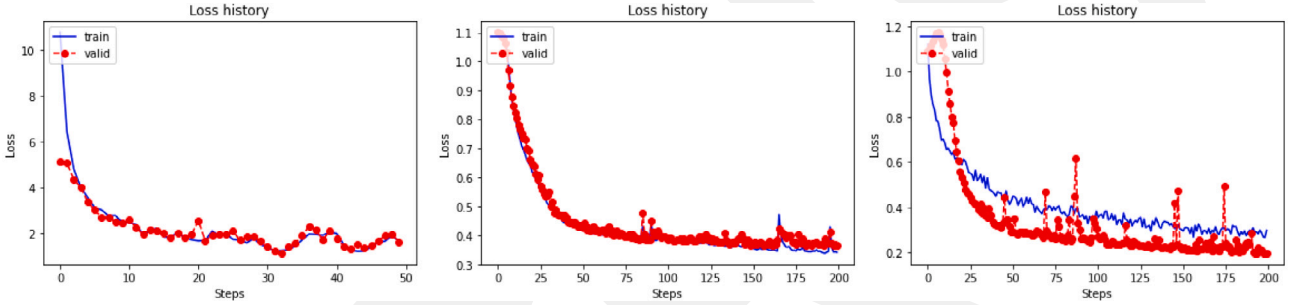


Fig. 6. The loss history of VGG16, LSTM, and GRU models on training and validation steps.

of validation sets are 1.1445, 0.3642, and 0.1970 after running 33, 162, and 200 steps on VGG16, LSTM, and GRU models, respectively. The validation loss in the GRU model is better than the training loss. The reason is that the GRU uses simpler gates and dropout is enabled during training, whereas it is not enabled during validation/testing on the Keras library.

5.2. Battery lifetime

In this study, while the vehicles are passing through 3-D magnetic sensor node, the vector magnitude-dependent measurement method is used to detect magnetic changes, and the time-dependent measurement method is used to sample the movement of the vehicle. Using a power analyzer named EnergyTrace, the sensor's current consumption is measured when it is in a sleep state as well as when it is in active state (when a vehicle passes by). In the sleep state, the current consumption is very low.

In general, the communication between the sensor and the gateway is the major cause of increased battery consumption. In this communication process, by using the data aggregation method, it is possible to reduce current consumption by selecting a proper packet size.

In addition, Table 7 shows the lifespan of the node versus the number of samples taken from the sensor node. Based on the current consumption characteristics of the magnetic sensor node, the proposed magnetic sensor node can operate for up to 2 years without requiring battery replacement, while taking 50 samples from the vehicle.

Table 7
Battery lifetime based on the number of samples taken from the vehicle.

# of samples taken from vehicle	Batter lifetime (day)
25	858
50	706
100	600
150	551
200	521
250	500
300	484

6. Conclusion

The classification of vehicles has evolved into an essential component of ITS to enhance the comfort of people, enhance traffic management, and pave the way for the further growth of transportation infrastructure. Significant investments are being made and used in the development, implementation, and maintenance of traffic monitoring systems. In particular, for effective traffic management on intermediate roads, it is important to have information about the types of vehicles. The developed system predicts the vehicle types with the help of deep learning models. To the best of our knowledge, this article is the first study focusing on a soft voting ensemble method with deep learning techniques for vehicle classification systems with a low-cost, battery-operated 3-D magnetic sensor node. Also, compared to other 3-D magnetic sensor studies, this paper is the first study focusing on energy efficiency. Comparative performance results show that the accuracy and f-measure of the proposed approach based on the soft voting ensemble method using deep learning classifiers are 92.92% and 93.42%, respectively, and the battery lifetime is up to 2 years. The main goal of this study is to achieve a reliable system that can produce satisfying results in any environmental conditions while keeping the performance stable. Future work includes increasing the dataset and increasing the number of classes. In addition, other model architectures

can also be implemented using transfer learning, such as EfficientNet, Inception, and NasNet.

CRediT authorship contribution statement

Burak Kolukisa: Conception and design of study, Analysis and/or interpretation of data, Writing – original draft, Writing – review & editing. **Veli Can Yildirim:** Acquisition of data, Writing – original draft. **Bahadir Elmas:** Analysis and/or interpretation of data. **Cem Ayyildiz:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data. **Vehbi Cagri Gungor:** Conception and design of study, Analysis and/or interpretation of data, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

This research was supported by the international funding agency EUREKA with the project name “NGA-ITMS (Next Generation Autonomous Intelligent Traffic Management System)". The project is funded nationally by TUBITAK TEYDEB with Project Number: 9180036.

All authors approved the version of the manuscript to be published.

References

[1] F.H. Somda, H. Cormerais, J. Buisson, Intelligent transportation systems: a safe, robust and comfortable strategy for longitudinal monitoring, IET Intell. Transp. Syst. 3 (2) (2009) 188–197, http://dx.doi.org/10.1049/iet-its:20080042.

[2] W. Ma, D. Xing, A. McKee, R. Bajwa, C. Flores, B. Fuller, P. Varaiya, A wireless accelerometer-based automatic vehicle classification prototype system, IEEE Trans. Intell. Transp. Syst. 15 (1) (2013) 104–111, http://dx.doi.org/10.1109/TITS.2013.2273488.

[3] H. Liu, J. Ma, T. Xu, W. Yan, L. Ma, X. Zhang, Vehicle detection and classification using distributed fiber optic acoustic sensing, IEEE Trans. Veh. Technol. 69 (2) (2019) 11363–11374, http://dx.doi.org/10.1109/TVT.2019.2962334.

[4] J. Chang, L. Wang, G. Meng, S. Xiang, C. Pan, Vision-based occlusion handling and vehicle classification for traffic surveillance systems, IEEE Intell. Transp. Syst. Mag. 10 (2) (2018) 80–92, http://dx.doi.org/10.1109/MITS.2018.2806619.

[5] E.H. Ng, S.L. Tan, J.G. Guzman, Road traffic monitoring using a wireless vehicle sensor network, in: International Symposium on Intelligent Signal Processing and Communications Systems, 2009, pp. 1–4, http://dx.doi.org/10.1109/ISPACS.2009.4806673.

[6] M.I. Asborno, C.G. Burris, S. Hernandez, Truck body-type classification using single-beam LiDAR sensors, Transp. Res. Rec. 2673 (1) (2019) 26–40, http://dx.doi.org/10.1177/0361198118821847.

[7] S. Meta, M.G. Cinsdikici, Vehicle-classification algorithm based on component analysis for single-loop inductive detector, IEEE Trans. Veh. Technol. 59 (6) (2010) 2795–2805, http://dx.doi.org/10.1109/TVT.2010.2049756.

[8] S.A. Rajab, A. Mayeli, H.H. Refai, Vehicle classification and accurate speed calculation using multi-element piezoelectric sensor, in: IEEE Intelligent Vehicles Symposium Proceedings, 2014, pp. 894–899, http://dx.doi.org/10.1109/IVS.2014.6856432.

[9] M. Stocker, M. Rönkkö, M. Kolehmainen, Situational knowledge representation for traffic observed by a pavement vibration sensor network, IEEE Trans. Intell. Transp. Syst. 15 (4) (2014) 1441–1450, http://dx.doi.org/10.1109/TITS.2013.2296697.

[10] M. Bottero, B. Dalla Chiara, F.P. Deflorio, Wireless sensor networks for traffic monitoring in a logistic centre, Transp. Res. C 26 (2013) 99–124, http://dx.doi.org/10.1016/j.trc.2012.06.008.

[11] S. Taghvaeeyan, R. Rajamani, Portable roadside sensors for vehicle counting, classification, and speed measurement, IEEE Trans. Intell. Transp. Syst. 15 (1) (2013) 73–83, http://dx.doi.org/10.1109/TITS.2013.2273876.

[12] B. Yang, Y. Lei, Vehicle detection and classification for low-speed congested traffic with anisotropic magnetoresistive sensor, IEEE Sens. J. 15 (2) (2014) 1132–11383, http://dx.doi.org/10.1109/JSEN.2014.2359014.

[13] F. Li, Z. Lv, Reliable vehicle type recognition based on information fusion in multiple sensor networks, Comput. Netw. 117 (2017) 76–84, http://dx.doi.org/10.1016/j.comnet.2017.02.013.

[14] W. Balid, H. Tafish, H.H. Refai, Intelligent vehicle counting and classification sensor for real-time traffic surveillance, IEEE Trans. Intell. Transp. Syst. 19 (6) (2017) 1784–1794, http://dx.doi.org/10.1109/TITS.2017.2741507.

[15] H. Dong, X. Wang, C. Zhang, R. He, L. Jia, Y. Qin, Improved robust vehicle detection and identification based on single magnetic sensor, IEEE Access 6 (2018) 5247–5255, http://dx.doi.org/10.1109/ACCESS.2018.2791446.

[16] C. Xu, Y. Wang, X. Bao, F. Li, Vehicle classification using an imbalanced dataset based on a single magnetic sensor, Sensors 18 (6) (2018) 1690, http://dx.doi.org/10.3390/s18061690.

[17] X. Zhang, H. Huang, Vehicle classification based on feature selection with anisotropic magnetoresistive sensor, IEEE Sens. J. 19 (21) (2019) 9976–9982, http://dx.doi.org/10.1109/JSEN.2019.2928828.

[18] X. Chen, X. Kong, M. Xu, K. Sandrasegaran, J. Zheng, Road vehicle detection and classification using magnetic field measurement, IEEE Access 7 (2019) 52622–52633, http://dx.doi.org/10.1109/ACCESS.2019.2908006.

[19] W. Li, Z. Liu, Y. Hui, L. Yang, R. Chen, X. Xiao, Vehicle classification and speed estimation based on a single magnetic sensor, IEEE Access 8 (2020) 126814–126824, http://dx.doi.org/10.1109/ACCESS.2020.3008483.

[20] S.C. Huang, B.H. Chen, Automatic moving object extraction through a real-world variable-bandwidth network for traffic monitoring systems, IEEE Trans. Ind. Electron. 61(4 (2013) 2099–2112, http://dx.doi.org/10.1109/TIE.2013.2262764.

[21] cc1312r data sheet, product information and support, URL http://www.ti.com/product/CC1312R.

[22] skyworks, URL https://www.skyworksinc.com.

[23] US Federal Highway Administration (FHWA), URL https://highways.dot.gov/.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[25] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. Artificial Intelligence Res. 16 (2002) 321–357, http://dx.doi.org/10.1613/jair.953.

[26] Transfer learning for deep learning, URL https://machinelearningmastery.com/transfer-learning-for-deep-learning/.

[27] F. Chollet, et al., Keras, GitHub, 2015, URL https://github.com/fchollet/keras.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[29] K. Greff, R.K. Srivastava, J. Koutník, B.R. Steunebrink, J. Schmidhuber, LSTM: A search space odyssey, IEEE Trans. Neural Netw. Learn. Syst. 28 (10) (2016) 2222–2232, http://dx.doi.org/10.1109/TNNLS.2016.2582924.

[30] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, 2014, arXiv, http://dx.doi.org/10.48550/arxiv.1406.1078.

[31] E. Baykal, H. Dogan, M.E. Ercin, S. Ersoz, M. Ekinci, Transfer learning with pre-trained deep convolutional neural networks for serous cell classification, Multimedia Tools Appl. 79 (21) (2020) 15593–15611, http://dx.doi.org/10.1007/s11042-019-07821-9.

[32] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2009) 1345–1359, http://dx.doi.org/10.1109/TKDE.2009.191.

[33] F. Chollet, Data Preprocessing, Keras, URL https://keras.io/api/preprocessing/image/.

Burak Kolukisa received the B.S. in Computer Engineering From Erciyes University, Kayseri, in 2016 and M.S. degrees in Electrical and Computer Engineering from Abdullah Gul University, Kayseri, Turkey, in 2020. Currently, he is working as a Research Assistant at the Department of Computer Engineering at Abdullah Gul University. His current research interests are Data Mining, Machine Learning, and Deep Learning. He is also a Ph.D. candidate in the Electrical and Computer Engineering program at Abdullah Gul University, Kayseri, Turkey.

Veli Can Yildirim received his B.S. degree in Control and Automation Engineering from Yildiz Technical University, Istanbul, in 2019. Currently, he is working as an Embedded Software Developer at GOHM Electronics and Embedded Intelligence Systems. His current research interests are Internet of things, machine to machine communications, and wireless ad hoc and sensor networks.

Bahadir Elmas received the B.S. in Electronics and Communications Engineering From Yildiz Technical University, Istanbul, in 1999, Master degree in Electronics from Yildiz Technical University, Istanbul, Turkey, in 2006 and Doctorate degrees in Mathematic from Mimar Sinan Fine Arts University, Istanbul, Turkey, in 2013. Currently, he is working as an Assistant Professor at the Department of Statistics, Mimar Sinan Fine Arts University. His current research interests are Data Mining, Machine Learning, and Deep Learning.

Cem Ayyildiz received his B.S. degree in Electrical and Electronics Engineering from Middle East Technical University, Ankara, Turkey in 2006 and M.S. degree in Microelectronics from Bremen University of Applied Sciences, Bremen, Germany in 2008. He received his Ph.D. degree in Computer Engineering program at Bahcesehir University, Istanbul, Turkey, in 2019. Currently, he is working at GOHM Electronics, Mugla, Turkey. His current research interests are wireless and mobile communications, and Internet of things.

V. Cagri Gungor received his B.S. and M.S. degrees in Electrical and Electronics Engineering from Middle East Technical University, Ankara, Turkey, in 2001 and 2003, respectively. He received his Ph.D. degree in electrical and computer engineering from the Broadband and Wireless Networking Laboratory, Georgia Institute of Technology, Atlanta, GA, USA, in 2007. Currently, he is a Professor and Chair of Computer Engineering Department, Abdullah Gul University, Kayseri, Turkey. His current research interests are smart grid communications, machine-to-machine communications, next-generation wireless networks, wireless ad hoc and sensor networks, cognitive radio networks.