Доклади на Българската академия на науките
Comptes rendus de l'Académie bulgare des Sciences
Tome 76, No 11, 2023

ENGINEERING SCIENCES
Control theory

SELF-DRIVING CAR CONTROL MODEL EXTENSION WITH VOICE COMMANDS CONTROL

Snezhana G. Pleshkova

Received on June 21, 2023
Presented by Ch. Roumenin, Member of BAS, on July 31, 2023

Abstract

The developed experimental models of self-driving car demonstrate high accuracy (about 99%), but there is still a need to improve the overall safety of real traffic on the roads. Especially when there are people in the car, if the autonomous vehicle loses control on the road, the quick intervention to prevent a possible crash or a more serious road accident can be only through voice commands between the person and the execution control devices of self-driving car. The existing self-driving car control models are mainly based on incoming from mounted on the autonomous vehicle video cameras information, processed from deep learning neural networks and artificial intelligence. This paper proposes to extend these models with voice commands recognition, spoken by a person in the car, in order to correct the self-driving car movement, to prevent possible traffic accidents, and therefore to increase traffic safety. For this purpose deep learning neural network with artificial intelligence is developed to recognize the spoken by the person voice commands, which can be interpreted by the executive control devices of the autonomous vehicle. The presented results from simulation tests show the ability of the proposed extended self-driving car control model to correct with voice commands the self-driving car motion on the road leading to essential increase of the safety of traffic.

Key words: autonomous vehicles, self-driving car control, voice commands control, CNN networks

Introduction. The self-driving cars are one of the most current real-world advances in artificial intelligence [1, 2]. The tested models of self-driving cars used

DOI:10.7546/CRABS.2023.11.12

deep learning neural networks to analyze the observed by video cameras road view [3]. The achieved about 99% accuracy of trained experimental models for exact following of the middle lane of the road [4] has stimulated the desire for mass application and production of autonomous vehicles, both as personal vehicles and even for buses, trucks and cargo vehicles [5]. In order this real perspective for future car transport to happen it is necessary to ensure a very high level of security against possible road accidents [6]. So, when the autonomous vehicle loses control and if there are persons in the self-driving car, it should be relied on quick human intervention to prevent a possible crash or a more serious road accident, especially in the initial periods of real use of autonomous vehicles on the road [7,8]. It can be assumed that under new traffic safety standards for autonomous vehicles, they will not be equipped with manual controls such as pedals and steering wheels [8]. Therefore, the only intervention to correct the self-driving car motion and to prevent a traffic accident can be the spoken voice commands from the person in the autonomous car. This defines the goal of this paper to extend the existing automatic visual models for control of autonomous vehicles with speech model using the recognition of person voice commands in order to correct the movement of the self-driving car. To realize this goal is proposed to create neural network with deep learning and artificial intelligence to recognize the person voice commands and interpret them from the executive devices of the autonomous vehicle. Simulation tests are carried out and experimental results are presented. These results show the ability of the proposed extended speech self-driving car control model to correct self-driving car motion on the road leading to essential increasing of the safety of traffic.

Block scheme of the proposed extension of self-driving car control model to correct self-driving car traffic with recognized human voice commands. Figure 1 presents the proposed extension for self-driving car control model with recognized human voice commands in critical traffic situations to prevent possible traffic accidents.

The block in the upper part of Fig. 1 represents the existing widespread simulation models for automatic self-driving car visual control [9]. If the person in the self-driving car notices a critical road situation, he can pronounce the appropriate words or whole sentences in microphone built in the proposed in Fig. 1 model. The audio signal from the microphone is used in the following two blocks: Voice commands recognition and Voice commands detection.

Voice commands recognition block in Fig. 1 performs human voice commands recognition using audio signal from the microphone. The results as recognized voice commands are used in the next block, shown in Fig. 1, to be interpreted and sent as execution operations via standard CAN (Controller Area Network) interface or (CAN bus) [10] to self-driving car control modules, shown in Fig. 1. The usually used execution operations to drive each car are: Brake, Throttle and Steer. Therefore, the issued by human in the car voice commands must

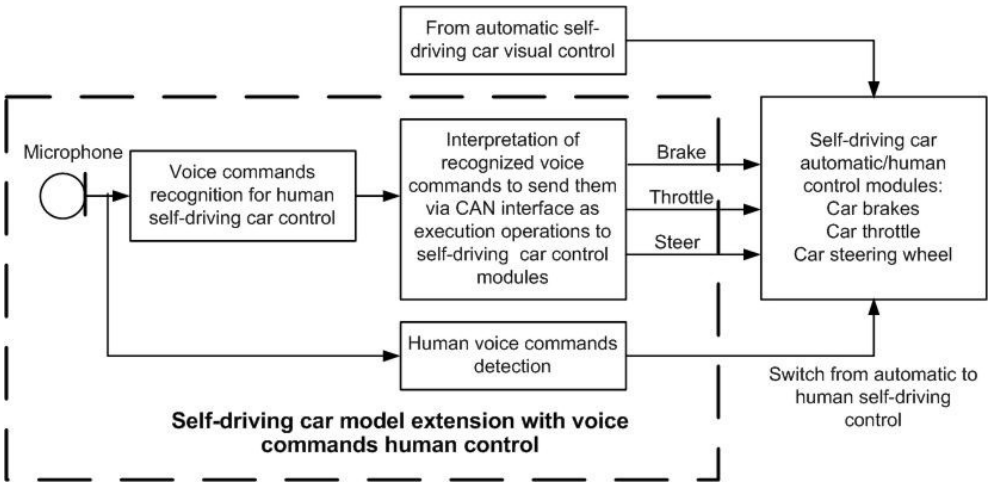

Fig. 1. Block scheme of the proposed extension for self-driving car control model with recognized voice commands in critical traffic situations to prevent possible traffic accidents

correspond to these tree mentioned operations executed by the self-driving car control modules. This means that between issued by the person in the car voice commands and their interpretation by the executive self-driving car control modules must exist the following correspondence: voice commands like "Stop" or "Hit the brakes" and "Release the brake" must be interpreted as "Brake" action; voice commands "Slow down the speed" and "Increase the speed" must be interpreted as "Throttle" action, respectively; voice commands "Turn left" and "Turn right" must be interpreted as "Steer" action.

Voice commands detection block, shown in Fig. 1, is intended to register the moments when the person in the self-driving car notices a critical road situation and pronounces the necessary voice command. The detected information for pronunciation of voice command, on the output of Voice commands detection block, shown in Fig. 1, is used to switch the self-drive car control modules working mode from automatic to human control.

From the block diagram in Fig. 1 it is possible to determine the following additional blocks necessary to be developed for the proposed extension with voice commands autonomous vehicle control model: voice commands recognition for self-driving car human control; voice commands detection to switch from automatic to human mode of the self-driving car control model with voice command; interpretation of the recognized voice commands to be sent via CAN interface as execution operations to self-driving car control modules.

The abovementioned additional blocks described in Fig. 1 are included in the proposed and presented below block algorithm of the self-driving car control model combining the automated visual control with human voice command control.

Block algorithm of the proposed self-driving car control model combining the automated visual control with human voice command control. The proposed block algorithm of self-driving car control model combining the automated visual control with human voice command control is presented in Fig. 2.

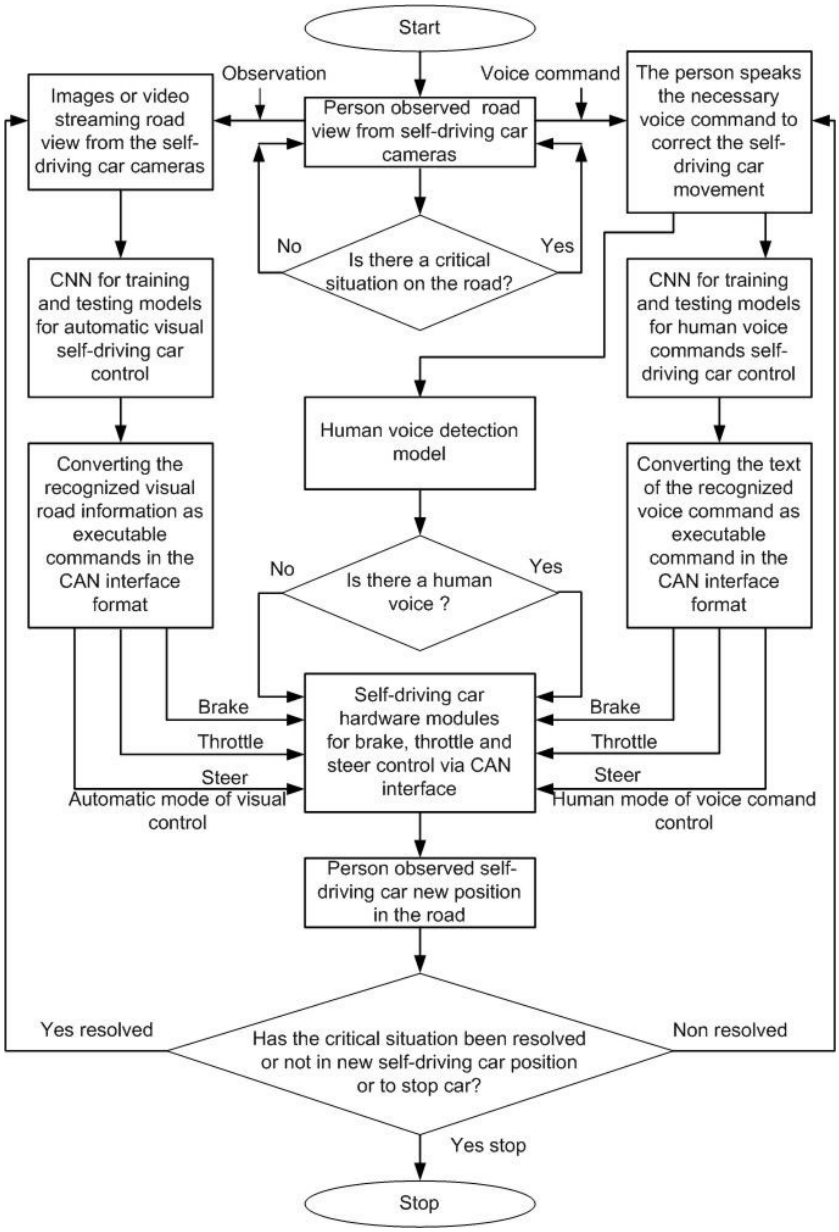

Fig. 2. Block algorithm of the proposed self-driving car control model combining the automated visual control with human voice command control

The block algorithm starts in automatic mode of self-driving car control model. This automatic mode utilizes information as images or video stream of road view from self-driving car cameras in block CNN (Convolutional Neural Network) for training and testing models for automatic visual self-driving car control. The results from this block convey the necessary information to make automatic changes of self-driving car position and orientation in concordance with the current road traffic situation. Next, these results are converted as executable commands in the CAN interface format to be used by self-driving car hardware modules for brake, throttle and steer control via CAN interface in automatic mode of visual self-driving car control. Meanwhile, the person observes the road view from self-driving car cameras. If the person does not notice critical situations, the self-driving car control model remains in automatic mode using the visual information for visual self-driving car control.

If the person notices a critical situation, he pronounces the necessary voice command to correct the self-driving car position and orientation and to resolve, if possible, the existing current critical situation. The pronounced voice command is recognized in block CNN network for training and testing the model for human voice commands self-driving car control. The result from this block, after training and testing, is text corresponding to spoken voice command. The text is converted as executable commands in the CAN interface format to be used by self-driving car hardware modules for brake, throttle and steer control via CAN interface in human mode of voice commands self-driving car control. To do this it is necessary to change self-driving car control mode to human mode switching also the mode of the block self-driving car hardware modules for brake, throttle and steer control via CAN interface. This switching can be done if the moment of appropriate voice command pronunciation by the person is known. Therefore, to do this the algorithm shown in Fig. 2 is proposed to add the block for human voice detection model used to switch correctly the mode from automatic visual self-driving car control to human with voice commands mode self-driving car control or vice versa. The person observing the self-driving car new position in the road after the implementation of pronounced voice command decides whether or not the critical road traffic situation is resolved. If in the new self-driving car position the critical situation has been resolved the self-driving car control mode is switched to automatic visual self-driving car control. Or if this situation is not resolved the self-driving car control remains in voice commands self-driving car control mode. Therefore, the person can continue to pronounce voice commands trying once again to correct the position and orientation of self-driving car to resolve the existing critical road traffic situation. Finally, if it is really not possible to resolve the existing critical road traffic situation, the person must stop the self-driving car to avoid the possible road accident.

Simulation of the proposed self-driving car control model combining the automated visual control with human voice command control. The block algorithm described above is simulative. Some of the blocks, related to the automatic visual control of the autonomous car are used from existing developments [11]. Therefore, in this paper only the blocks related to the human voice commands for self-driving car control are simulated. One of the most important of these blocks is the block for creating, training and testing CNN network to recognize human voice commands. This block is simulated as a modification of CNN network model developed in Python TensorFlow library [12] and is presented briefly in this paper as a sequence of Python pseudo code functions in Table 1.

The presented in Table 1 Python pseudo code can be described more clearly with the following additional explanations. The created data set is based on the existing voice command data set [12] containing general set of frequently used voice commands and it is extended with additional voice commands specific for self-driving car control like "stop", "turn left", "turn right", etc. All variants of the different (total 1000) utterances, spoken by an equal number of men and women for each voice command, are divided into two parts for use in training 70% and for testing 30%.

The short-time Fourier transform (STFT) [13] spectrograms are chosen as the recognition features extracted from voice commands and are the input voice information in CNN network for voice commands recognition. Mel-frequency cepstral coefficients (MFCC) [14], are also commonly used for voice recognition, but using spectrograms can reduce the time for calculation.

The created CNN network model begins with input layer chosen as full connected layer receiving voice command spectrograms as two dimensional matrix. It is followed by first 2D convolution layer with the following parameters: input with 32×32 size of resized matrix of spectrograms, 3×3 kernel dimension and type of activation "relu", i.e. rectified linear unit. The second 2D convolution layer is with similar parameters, except input size as 64×64. After these two 2D convolutional layers are added the existing in each CNN network layers like MaxPooling, Dropout, Flatten and Dense layers necessary to non-linear transformation and 2D to 1D dimension transformation of values on output of second CNN network. The simulation of human voice command model of a self-driving car control is tested and the accuracy of 0.8913 or about 89% is achieved, presented below as output printed lines in Python's Integrated Development and Learning Environment – IDLE [15] after training and testing of the simulated CNN network model with 100 epochs:

100/100 [==========] – 0s 7ms/step – loss: 0.0120 – accuracy:0.8913

{'loss': 0.011960983276372, 'accuracy': 0.8913461446762085}.

Of course, it is possible to extend this accuracy including additional samples of voice command in data set and extend the number of epochs in the training, but

Table 1

The Python pseudo code of CNN network model for voice commands recognition

Action description	Python pseudo code	Commentaries
Create self-driving car control voice commands dataset	tf.keras.utils.get_file('mini_speech_commands.zip', origin="http://storage.googleapis.com/ download.tensorflow.org/data/ mini_speech_commands.zip"	Self-driving car voice commands are added to the existing voice command dataset
Convert voice commands in dataset to spectrograms	def get_spectrogram(waveform): # Convert the waveform to a spectrogram via a STFT. spectrogram = tf.signal.stft(waveform, frame_length=255, frame_step=128)	Using Short time Fourier Transform (STFT) to calculate voice commands features as spectrogram
Build CNN model for voice commands recognition	model = models.Sequential([layers.Input(shape=input_shape), # Downsample the input. layers.Resizing(32, 32), # Normalize. norm_layer, layers.Conv2D(32, 3, activation='relu'), layers.Conv2D(64, 3, activation='relu'), layers.MaxPooling2D(), layers.Dropout(0.25), layers.Flatten(), layers.Dense(128, activation='relu'), layers.Dropout(0.5), layers.Dense(num_labels),])	CNN model consists of: input layer receiving voice commands spectrograms; 2D convolution layers with corresponding parameters; accompanying layers like MaxPooling, Dropout, Flatten and Dense
Train CNN model over chosen numbers of trained epochs for example: 100 EPOCH	EPOCHS = 100 history = model.fit(train_spectrogram_ds, validation_data=val_spectrogram_ds, epochs=EPOCHS, callbacks=tf.keras.callbacks.EarlyStopping (verbose=1, patience=2),)	Start CNN model in training mode with chosen epochs number to achieve the desired accuracy
Test CNN model after training to evaluate the achieved accuracy	model.evaluate(test_spectrogram_ds, return_dict=True)	Start CNN model in test mode
Save the trained CNN model	tf.saved_model.save(export, "saved")	Return to training step to do additional training epochs if desired accuracy is not achieved or save model and stop CNN training.

it is more appropriate to do the tests for determining the ability of the proposed extension of self-driving car control model with human voice command control using the achieved current 89% accuracy. The tests for this purpose are carried out integrating the trained CNN network in the Automated Driving Toolbox in Matlab [16] as a part of Matlab Advanced driver-assistance systems – ADAS [17], because of their capabilities to visualize road maps imported as data maps from HERE HD Live Map [18] and OpenDRIVE road networks [19]. Also the existing in Fig. 2 block Self-driving car hardware modules for brake, throttle and steer control via CAN interface is simulated as Matlab Simulink Vehicle with Four-Wheel Drive Model [20]. Figure 3 presents a visualization of the achieved results demonstrating the ability to correct, in case of critical situation, the self-driving car direction by voice command pronounced by the person observing the road map.

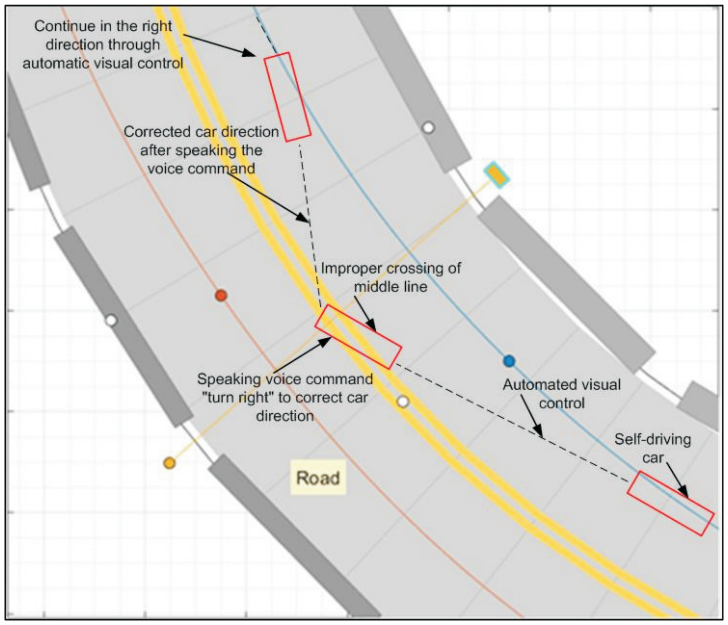

Fig. 3. Visual representation of the ability to correct in critical situations the self-driving car direction by voice command

The visualization in Fig. 3 shows a fragment of the simulated road map with the starting position of the autonomous vehicle (indicated by a red rectangle) at the bottom right of the figure and in autonomous visual control mode to follow the middle lane of the road. The human observes on simulated road map the autonomous car's movement in automatic visual mode and notices that the car is steering incorrectly to cross the middle lane of the road. He immediately speaks the necessary voice command "turn right" into the microphone of the simulation stage. The voice command is detected and driving control mode of the

autonomous car is switched from automatic visual to human control. Therefore, the spoken voice command "turn right" is executed and interpreted by the self-drive car control modules as "steer" action, which is indicated on simulated road map in Fig. 4 as corrected car direction after speaking the voice command. After that the self-driving car control is switched to automatic visual mode and new direction in the road is set correctly to follow the middle lane of the road.

Conclusion. An extension for voice commands control of existing only with automatic visual control models of self-driving car control is proposed. The extension is included as an appropriate part to the existing model only with automated visual control of self-driving car. Voice command control of self-driving car must have the priority in critical road traffic situations. That is why, in the block scheme developed in the paper it is proposed immediately after speaking the voice command, the automatic visual control mode of the car to be turned off and recognized spoken voice command to be interpreted and transmitted through the CAN interface for immediate execution by the schemetically realized executive control mechanisms of the car. All these actions are implemented in the algorithm developed and described in the paper. A software application of a model for voice control of an autonomous car has been developed based on the proposed algorithm. In the developed software application, a high-accuracy of about 89% CNN neural network for voice command recognition was created and trained. The trained CNN network is applied to recognizing the spoken voice commands in a model of road traffic created for simulating in experiments the combined control with voice commands and with autonomous visual control of the self-driving car. The results of the numerous experiments carried out are summarized by the presented in Fig. 4 simulated visual part of a road map. There are simulated the suitable road traffic situations for movement of self-driving car: normal non critical situations of following the middle lane of the road and critical situation of crossing the middle lane of the road by self-driving car. From the achieved results it is evident that the simulated critical situation of crossing middle line of the road by self-driving car is exactly corrected when the human observing critical situation, speaking the necessary voice command and self-driving car changing it direction following correctly the middle lane of the road. Also the mode of self-driving car control is switched correctly from automated visual to human voice control at the moment of the spoken voice command and after correction of self-driving car direction the mode of self-driving car control is returned back to automated visual mode.

It can be summarized that the proposed extension of the model for automatic visual self-driving cars control can by means of voice commands quickly, efficiently and with sufficient accuracy control the movement of self-driving cars which is important for avoiding critical or fatal traffic situations. This is confirmed by the briefly presented through simulations experimental results for the accurate correction of the movement of the self-driving car, in the event of an incorrect crossing of the middle lane of the road.

The results achieved and confirmed by simulations are a real reason to use them in future research and practical implementations testing on real road situations of the proposed in this paper extension of automatic visual control model for self-driving car control with voice commands control.

REFERENCES

[1] Self-driving cars Technology, Solutions for self-driving cars, NVIDIA DRIVE End-to-End Platform for Software-Defined Vehicles, https://www.nvidia.com/en-us/self-driving-cars/.

[2] Pisarov J., G. Mester (2020) The Future of Autonomous Vehicles, FME Transactions, **49**(1), 29–35.

[3] Razzaq W., U. Arif, Z. M. U Din (2021) Visual Perception Deep Drive Model for Self-Driving Car, Pakistan J. Sci. Res., **1**(1), 18–21, https://doi.org/10.57041/pjosr.v1i1.5.

[4] Wu Z., K. Qiu, T. Yuan, H. Chen (2021) A method to keep autonomous vehicles steadily drive based on lane detection, International Journal of Advanced Robotic Systems, **18**(2), 172988142110029.

[5] Autonomous driving trucks: an efficient future? https://traton.com/en/newsroom/current-topics/autonomous-driving-trucks-an-efficient-future.html.

[6] European Commission, Autonomous Vehicles & Traffic Safety, 2018. https://road-safety.transport.ec.europa.eu/system/files/2021-07/ersosynthesis2018-autonomoussafety.pdf.

[7] Connected & Automated Mobility 2025: Realising the benefits of self-driving vehicles in the UK, Presented to Parliament by the Secretary of State for Transport and the Secretary of State for Business, Energy and Industrial Strategy by Command of Her Majesty, August 2022.

[8] Dimitrov K., T. Valkovski, I. Damyanov, G. Mladenov (2022) Petrol and Diesel Engines Sound Measuring and Analyzing in Real Road Conditions, 30th National Conference with International Participation (TELECOM), Sofia, Bulgaria, 1–4, doi: 10.1109/TELECOM56127.2022.10017327.

[9] Calderon P. B., K. H. Okabe, F. M. Moreno, J. M. Ugarte Yaffar (2021) Autonomous Driving on Nvidia Dave-2. In: Selected Topics in Artificial Intelligence from Professor Gissel Velarde.

[10] CAN bus the ultimate Guide, CSS Electronics, Jan 31, 2023, www.csselectronics.com.

[11] Bojarski M., B. Firner, B. Flepp, L. Jackel, U. Muller et al. (2016) End-to-End Deep Learning for Self-Driving Cars, NVIDIA Technical Block. https://developer.nvidia.com/blog/deep-learning-self-driving-cars/.

[12] Tensorflow Speech Recognition Challenge. https://www.kaggle.com/competitions/tensorflow-speech-recognition-challenge/.

[13] Short-time processing of speech signals. https://speechprocessingbook.aalto.fi/Representations/Short-time{_}processing.html.

[14] Mistry D. S., A. V. Kulkarni (2013) IJERT-Overview: Speech Recognition

Technology, Mel-frequency Cepstral Coefficients (MFCC), Artificial Neural Network (ANN), International Journal of Engineering Research & Technology (IJERT), **2**(10), ISSN: 2278-0181.
[15] Python 3.11.4 Version. https://www.python.org/.
[16] Automated Driving Toolbox in Matlab. https://www.mathworks.com/products/automated-driving.html.
[17] Advanced driver-assistance systems. https://www.mathworks.com/discovery/adas.html.
[18] HERE HD Live Map. https://www.here.com/platform/HD-live-map.
[19] OpenDRIVE road networks. https://www.asam.net/standards/detail/opendrive/.
[20] Matlab Simulink Vehicle with Four-Wheel Drive Model. https://www.mathworks.com/help/sdl/ug/vehicle-with-four-wheel-drive.html.

Faculty of Telecommunications
Technical University of Sofia
8 Kliment Ohridski Blvd
1000 Sofia, Bulgaria
e-mail: snegpl@tu-sofia.bg