Michał **BUGAŁA**

ALGORITHMS APPLIED IN AUTONOMOUS VEHICLE SYSTEMS

Abstract. Many research centres in the world that deal with the problems of the manufacture of land vehicles, especially those intended for transport and communication in urban traffic, are still working on the development of a vehicle equipped with systems that do not require human participation in the process of driving a vehicle. The goals to be pursued are to ensure maximum safety (minimize accidents involving people) and to optimize transport costs (eliminating the driver from the vehicle, optimal route selection). This article, which is the result of broad studies conducted at OBRUM, discusses the development paths of autonomous vehicles and key algorithms of autonomous vehicles described in detail in the literature cited. The presented results constitute a starting point for further work on an autonomous vehicle to be carried out at OBRUM.

Keywords: algorithmics, autonomy, autonomous vehicle, autonomous vehicle system.

1. INTRODUCTION

The article presents and discusses basic algorithms applied in autonomous vehicle systems (land vehicles), the use of which may vary depending on the assumed level of autonomy and the sensors used. The most important tasks of autonomous vehicle systems, regardless of the level of automation and the type of sensors, include collecting information about the environment, estimating their position in the environment, predicting the movement of other objects and sending calculations made to the vehicle control systems (steering wheel, throttle, brake, etc.). The material presented in the article is divided into two parts: historical (section 2) and solutions currently applied (section 3).

2. HISTORICAL SKETCH

A debatable, or perhaps even controversial issue may be the need to implement autonomous vehicle systems in the context of passenger cars, their users and forthcoming car rental services [1]. History shows that such attempts have already been made after the appearance of the first man-operated vehicles. The above doubts do not refer, of course, to military solutions [2] and the concepts of autonomous urban transport systems [3], [4].

2.1. First attempts on control automation

The history of the first attempts to automate vehicle control dates back to the 1920s, when in 1926 the Chandler company presented a vehicle radio-controlled by an operator sitting in another vehicle following the unmanned vehicle. This, of course, was not a case of vehicle autonomy, but it was the first time a car was moving on the road without a crew [5], [6]. In 1956 General Motors equipped an experimental Firebird II model with a receiver of guiding signals transmitted from a wire embedded in the roadway of the so-called *highway of the future* [7]. In 1979 an experimental Stanford Cart was able to move in closed space without human intervention using an image processing algorithm called The Cart's Vision Algorithm. This algorithm was inspired by the Blocks World planning method and it consisted in the reduction of the image to a set of edges, but it proved unsuitable for outdoor use where

Michał **BUGAŁA** – Ośrodek Badawczo-Rozwojowy Urządzeń Mechanicznych „OBRUM" sp z o.o., Gliwice

many complex shapes and colours existed [8]. However, the 1960 Blocks World method is today one of the best known methods in the field of artificial intelligence planning [9] and environment recognition [10], [11].

2.2. The first autonomous vehicle

In 1995, the experimental VaMP vehicle travelled over a thousand kilometres without human assistance and it was the first autonomous car able to move around a specific area. The prototype was able to drive in traffic and to overtake vehicles, and the implemented EMS-Vision autonomy system was based on data acquired from bifocal camera systems (45 and 15 degrees) mounted on biaxial platforms [12]. Such a selection of cameras (more recent versions of the EMS-Vision system use a system of three cameras [13]) was dictated by high-speed driving, observation of potential obstacles at larger and smaller distances, terrain unevenness and the ability to interpret the spatial layout. EMS-Vision made use of the following:

- road network map;
- static objects on the map that were used as landmarks (waypoints);
- statistical data (e.g. traffic lane width).

Acquiring dynamic knowledge, i.e. collecting information about the environment while driving, consisted in identifying objects (markers), calculating the positions of objects (location and orientation) in space using the HTC (Homogeneous Coordinate Transformation) algorithm and creating a scene tree containing these objects (the elements connecting objects in the scene tree were their relative positions). The system additionally had decision-making units for maintaining important objects in the field of vision (to be analyzed by specialized modules), for steering the vehicle, and a central decision-making unit. The central unit had a decision-making priority in the absence of tasks, conflicts between the control unit and the attention keeping unit, and also in the case of other problems. Specialized modules included in EMS-Vision:

- road recognition - the algorithm not only recognized the way, but created its model from linked segments and on this basis determined the position of the vehicle;
- attention control – responsible for communication with a two-axis rotary head that performs jump (saccadic) manoeuvres related to object tracking;
- navigation - the module calculated several routes to the destination along with the start and end time of the travel, the length of the route and the estimated travel ability (planning phase). The calculation results were sent to the central unit, which decided on the route selection. The selected route was added to the task list in the mission plan. The mission plan included tasks such as the selected route, tracking objects (road, landmarks) and other. It was possible to change the original mission plan during the mission, for example when an expected intersection could not be found;
- vehicle control (steering) - the module was responsible for analyzing information coming from decision-making units and performing algorithms for feedback control, feedforward control and transitional rules. The module evaluated the resulting sets of equations, calculated the corrective variables and controlled the actuators [14].

The above description of the EMS-Vision system does not include modifications [13], [15], which were created after 1995, but it illustrates the degree of complexity and the number of algorithms used, meant as defined actions necessary to perform specific tasks in autonomous vehicle systems.

2.3. DARPA Grand Challenge

Of great importance in the development of autonomous vehicle systems was a competition sponsored by DARPA (Defense Advanced Research Projects Agency), in which participants had to construct a vehicle that could drive 241 km (150 miles) across the Nevada desert. In 2004, none of the teams completed the task, and the best team travelled only 13 kilometres (8 miles). The problem was related to image analysis and the difficulty of recognizing the environment in which there are no clear lines (e.g. roadside with a pavement), and the difficulty of interpreting a large number of shadows on irregular desert terrain. Standard algorithms of image analysis (creating contours, matching lines, etc.), object identification and strategies to bypass obstacles did not work in 2004. In 2005, the use of machine learning was started in image analysis processes and five vehicles managed to complete the route [16].

The use of machine learning in image analysis gave rise to a new trend in building systems for autonomous vehicles. Major companies began building their own intelligent systems for autonomous vehicles, e.g. the Waymo project launched in 2009 by Google (members of the Stanley team who won the DARPA Grand Challenge in 2005 also participated in the project [17]). The system software of the Stanley vehicle contained 100,000 lines of code and was responsible for interpreting sensor data and making navigation decisions [18].

2.4. DARPA Urban Grand Challenge

In 2007, an urban competition was held in Victorville, California, with the participation of 53 teams. 11 teams made it to the final, of which 6 managed to complete the route. The software of the competition winner, the Tartan Racing team [19], consisted of 500,000 lines of code and allowed unattended navigation in the city street traffic. The Boss vehicle (Chevrolet Tahoe with hardware and software) used perceptual, planning and behavioural software to infer traffic conditions and make decisions while moving to the destination. The vehicle was equipped with a number of lasers, cameras and radars, which allowed it to plan the route taking into account static and dynamic obstacles (moving objects). Information on the environment was gathered by algorithms for finding and recognizing lane restrictions, parking limits, road signs and more. In addition, algorithms have been implemented to identify the dangerous behaviour of other drivers. The most important features of the technology developed by the Tartan Racing team include:

- driving in accordance with the road traffic rules (taking into account the priority rules of crossing intersections);
- detection and tracking other vehicles over long distances;
- finding parking spaces and parking itself;
- maintaining safe distance to followed vehicles;
- reacting to non-standard events (e.g. traffic jams) [20].

2.5. The first autonomous vehicles approved for road traffic

In 2014, Tesla launched the S model with software called *Autopilot Firmware 7.0*, which offered the second level of automation [21]. The description of the automation levels according to SAE International (previously SAE - *Society of Automotive Engineers*) is presented in Table 1.

Table 1. Automation levels according to SAE [22]

Level	Name	Description
0	No Automation	Complete lack of automation. All decisions and actions are taken by the driver.
1	Driver Assistance	Driver assistance systems, for example in accelerating/decelerating, based on external information. This level requirements are met by ACC (Adaptive Cruise Control), Parking Assistance and Lane Keeping Assistance (LKA).
2	Partial Automation	The systems of level 1 are assumed autonomous, but the driver must monitor the environment continuously.
3	Conditional Automation	The vehicle is driven by automatic systems, but the driver is obliged to react when he/she considers the decision of the automatic system to be incorrect.
4	High Automation	The vehicle is driven by automatic systems even when the driver does not respond appropriately in a dubious situation.
5	Full Automation	Full autonomy. The vehicle is able to perform all driving tasks under any conditions. The driver may still be able to control the vehicle.

It certainly was not a fully autonomous vehicle (model S) yet, but the then version of the software allowed autonomous driving along the road, changing lanes and parking on demand. The S model was one of the first autonomous cars available on the market (not fully autonomous, however, it aroused discussions about the manufacture and use of autonomous vehicles) [23].

Other manufacturers, the few to offer automation in their cars at that time, were: Ford (lane keeping assistance, automatic parallel parking, accident avoidance) [24], Mercedes [25], Audi and Nissan [26].

Currently, there is a race between passenger car manufacturers in providing their customers with the safest and, where possible, maintenance-free products. High competitiveness leads to limited access to information and the inability to accurately analyze the systems used in practice by car manufacturers.

3. CURRENT SOLUTIONS

This part of the article will present basic algorithms of image analysis, information storage and decision algorithms, which are used in the construction of prototype autonomous systems (not necessarily different from commercial solutions).

The number of events that may occur on the road, image diversity (landscape, season, time of day), different traffic rules in different countries (e.g. left-hand or right-hand traffic, road signs) requires a lot of knowledge and a number of proven solutions for the development of software for a fully autonomous vehicle. It is no exaggeration to say that every image processing and analysis algorithm can be used in autonomous vehicle systems. However, the number of algorithms used is limited by computational capabilities and is determined by the type of sensors used: camera, LASER, GPS, Infrared, LIDAR, etc. The algorithms are reviewed here in the following respects:

- environment recognition with the help of camera image (section 3.1):
  - traffic lane (road) recognition;
  - semantic image segmentation and identification of objects;
  - image based environment mapping (visual SLAM);
- environment recognition with the help of sensors (section 3.2):
  - sensor data based environment mapping (SLAM);
- detection and tracking of moving objects (DATMO, section 3.3);
- planning and decision-making (section 3.4);
  - route planning algorithms;
  - methods of planning and control of vehicle movement;
  - decision making mechanisms;
  - artificial intelligence.

3.1. Environment recognition with the help of camera image

Image from cameras is the richest and indispensable source of information in the process of environment identification. In addition to cameras, radar sensors (insensitive to weather, dust and pollution) and more expensive laser sensors - lidars are used [27].

3.1.1. Traffic lane (road) recognition

Convergence of parallel lines at one point in three-dimensional space constitutes valuable information in road identification, because the parallel lines that form the lane meet at one point in a two-dimensional image (image from a camera). Identification of a lane on the road consists in finding lines that converge at one point.

The VP (Vanishing Point) method for finding road in a desert, proposed by C. Rasmussen [28], consists in calculating the dominant orientations in image segments (image resolution 640 x 480 pixels divided into 72 segments), estimating the position of the point of convergence and tracking that point in subsequent image frames. The Rasmussen method, developed to recognize the road in a desert area, today plays an important role in detecting road lanes in an image [29] and forms the basis for more advanced algorithms [30].

3.1.2. Semantic image segmentation and identification of objects

Modern methods of image analysis, which make use of machine learning (machine vision), enable detecting specific categories of objects, including people, road signs and cars in complex images.

Machine learning can be used for semantic segmentation of images (e.g. FRNN (Full Resolution Residual Networks) method [31]) and for detailed description of an image, i.e. segmentation with an exact outline of objects. However, due to the large computational cost and the numerous artefacts created, image segmentation methods are not applied in practice [27], [32], [33], [34]. This approach may change with the development of dedicated equipment [35] [36] and of new, faster methods of semantic segmentation [33] [37].

Machine vision is used to detect specific (predefined) image features: detect traffic lanes, obstacles, moving objects and to estimate distance [27]. Specialized Advanced Driver Assistance Systems (ADAS) are able to identify road signs, traffic lights and markings on the road. More effective methods of machine learning to recognize road signs (and other image elements), along with the algorithms used, which appeared in various studies and were described in scientific articles, can be divided into 3 groups [34]:

- general algorithms of machine learning:
  - Viola-Jones method of 2001 [38];
  - SVM (Support Vector Machine) methods preceded by feature extraction algorithms, such as:
    - Histograms Of Gradients (HOG) [39],
    - Pyramid Histograms Of Gradients (PHOG) [40];
    - method that enables constructing a strong classifier from a large number of weaker classifiers – AdaBoost (Adaptive Boosting) [41];
    - Haar-like feature classifier [39];
    - feature extractor MSER (Maximally Stable Extremal Regions) [42];
    - HOG with MSER WaDe algorithms for region extraction [43];
    - HOG with LDA (Linear Discriminant Analysis) [44][45];
    - HOG with BCT (Bilateral Chinese Transform) and VBT (Vertex Bisector Transform) [46];
  - AdaBoost methods combined with:
    - feature detector LBP (Local Binary Pattern) [47];
    - image pyramid and HOG [48];
    - Haar-like classifier [49];
  - other:
    - methods that use genetic algorithms (K. Kaplan) [50];
    - BoVW (Bag of Visual Words) with SIFT (Scale-Invariant Feature Transform) algorithm [51];
- artificial neural networks:
  - VG-RAM-WNN (Virtual Generalizing Random Access Memory Weightless Neural Networks) [52];
  - PNN (Probabilistic Neural Networks) with CPT (Central Projection Transformation) and ACIS (Adaptive Colour Image Segmentation) [53];
- deep machine learning:
  - CNN (Convolutional Neural Network) [54];
  - CNN preceded by grey scale image processing by SVM [55];
  - use of two CNN networks to detect signs in an image and to identify the signs [56];

- CNN with RPN (Region Proposal Network) algorithm [57] in a Fast R-CNN (Fast Region-Based Convolutional Neural Network) [58];
- Faster R-CNN (Faster Region-Based Convolutional Neural Network) methods for detecting objects within the area of interest, including road signs [59].

According to [34] the most popular and most tested methods used for road sign recognition are the SVM methods. AdaBoost is characterized by fast execution time, and neural network methods are the slowest. Deep machine learning methods have demanding hardware requirements and can therefore be difficult to apply in ADAS systems. Faster R-CNN may be an alternative for the semantic image segmentation.

3.1.3. Image based environment mapping (visual SLAM)

SLAM (Simultaneous Localization And Mapping) enables precise determination of user's own position relative to the environment. In SLAM both the trajectory of the platform and the location of all landmarks are estimated on-line without the need for any a priori knowledge of location [60]. In the case of algorithms that use images, in which landmarks must be identified and tracked in every frame of the camera image, the requirements set for the equipment used (computing power) are very stringent. A three-dimensional map of the environment is built based on the landmarks, and the SLAM method itself is in the form of algorithms:

- EKF-SLAM (EKF - *Extended Kalman Filter*) algorithm of 1987, which makes use of the Kalman filter as the main estimator of environment structure [61];
- FastSLAM algorithm which uses RBPFs (Rao-Blackwellised Particle Filters) [62], [63];
- FastSLAM 2.0, a hybrid solution with combined use of Kalman filter and particle filters [64];
- RGB-D SLAM which utilizes image and image depth (Kinect) [65];
- ORB-SLAM and ORB-SLAM2 for single images, stereo images (stereoscopic vision) and RGB-D cameras [66]. The algorithm utilizes the ORB (Oriented FAST and Rotated BRIEF) descriptor [67];
- a system based on computational models of navigational processes in the hippocampus (part of the mammalian brain) called RatSLAM [68];
- LSD-SLAM – an algorithm that generates depth maps from individual image frames without using landmarks [69];
- L-SLAM – an algorithm that reduces the dimensionality of FastSLAM algorithms [70].

Visual SLAM methods are used primarily in vehicles of small dimensions, e.g. drones (example source: [71]).

3.2. Environment recognition with the help of sensors

RADAR (Radio Detection And Ranging) and LIDAR (Light Detection and Ranging) are the basic sensors used in practice for recognizing the environment [32], [72], [73]. Radar has been used in the automotive industry for a long time for the determination of speed, range and orientation of objects. It is not sensitive to weather conditions. Lidar, which is heavier and more expensive, but provides more accurate results (measurements), started to be used on a larger scale in 2007 (DARPA Urban Challenge) [74]. The use of radar and lidar is associated with environment mapping and SLAM algorithms, which are not burdened with the problem of identifying landmarks in the image.

A less advanced and, therefore, cheaper group of devices are ultrasonic sensors [75], mounted on, for example, car bumpers (parking sensors), which allow measuring distance to an obstacle.

3.2.1. Sensor data based environment mapping (SLAM)

SLAM algorithms can be classified according to the type of sensor:

- Radar:
  - GraphSLAM offline algorithm [76], based on constructing graphs and mapping environmental grid [77];
  - Graph-Based SLAM based on constructing graphs and finding node configurations of minimal error [78];
  - Real-Time Radar SLAM, which utilizes FastSLAM and GraphSLAM, and which is performed within 45 milliseconds (Intel Core i7-3830K) [79].
- Lidar:
  - ML-SLAM (ML – *Maximum Likelihood*) based on maximum likelihood estimation [80];
  - Credibilist SLAM (or C-SLAM) [82] based on TBM (*Transferable Belief Model*) [81];
  - ICP-SLAM [84] which utilizes ICP (*Iteractive Closest Point*) method of registering three-dimensional shapes [83];
  - Google's Cartographer SLAM, which utilizes, in addition to a lidar, IMU (*Inertial Measurement Unit*) and images from cameras [85], [86].

The LOAM algorithm (Lidar Odometry And Mapping), which consists in using distance measurements made by a biaxial lidar moving in six degrees of freedom, is also used for environment mapping [87].

3.3. Detection and tracking of moving objects (DATMO)

The DATMO (Detection And Tracking of Moving Objects) algorithms may be applied when images from a camera [88], radars [89] and lidars [90] are used.

When using the DATMO and SLAM methods, the benefits are mutual. The identification of dynamic objects in the environment and ignoring them by SLAM algorithms is of fundamental importance in precise mapping of urban areas with a large number of moving objects. On the other hand, the possibility of using the SLAM algorithm when tracking moving objects allows to better estimate global speeds and positions, and thus to better estimate the trajectories of moving objects [91]. A list of solutions with combined use of SLAM and DATMO, based on [92], is given in Table 2.

Table 2. SLAM and DATMO methods

Author(s)	SLAM Method	DATMO Method		Environment	Objects
		Data relations	Tracking		
Wang 2002-2004 [93]	Grid-based (EKF)	MHT (Multiple Hypothesis Tracking) [94]	IMM (Interactive Multiple Model) [95]	Open space	Humans, cars, bicycles, buses
Hähnel, Schulz, Burgard 2003 [96]	Grid-based (Bayes filter)	SBJPDA (Sample-based Joint Probabilistic Data Association Filters) [97]		Open space	Humans
Montesano, Minguez, Montano 2005 [98]	Grid map	NNR (Nearest Neighbor Rule) [99]	EKF	Room	Humans, doors
Solà 2007 [100]	BiCamSLAM (two cameras)	EKF		Room	Humans, boxes, tables, baskets
Vu 2009 [91]	ML-SLAM and EKF	DDMCMC (Data-Driven Markov Chain Monte Carlo)		Open space	Humans, cars, bicycles, buses
Vu, Burlet, Aycard 2010 [101]	Bayes, ICP	MHT	IMM	Open space	Humans, cars

DATMO methods of object detection and tracking developed independently, i.e. without employing SLAM methods, with the use of various sensors and in various configurations, are the subject of many studies, e.g.

- RoadPlot-DATMO with radars, 2D lidars and 3D lidar [102];
- the algorithm described in 2016 by F. J. Both [103] which utilizes GIW-PHD (Gaussian Inverse Wishart - Probability Hypothesis Density), a radar sensor and two cameras.

3.4. Planning and decision-making

Identification of unusual events on the road that can occur almost always and everywhere and can create potential problems must be resolved in a safe manner. Numerous studies have focused on ethical considerations, in cases where passengers' (or, in the case of incomplete autonomy, operator's) and other traffic participants' health is endangered, and on autonomous decisions in such cases [104], [105], [106].

3.4.1. Route planning algorithms

One of the first decisions to be taken by an autonomous vehicle is to choose a route (path) to the designated point, most often according to the criterion of distance, travel time or assumed fuel consumption. Heuristic algorithms, which take into account the criterion of travel, i.e. the cost on the node-connecting edges (weighted graph) in directed graphs, include:

- Bellman-Ford algorithm, with the restrictions in the form of the need to define edges of non-negative weights [107], [108];
- Dijkstra algorithm, which can be applied if the topology of roads is known [109];
- A* algorithm of 1968 [110] which, along with its modifications [111], is the solution most often applied [112], [113].

To locate your own vehicle and to determine the route on the map (known a priori), GPS sensors and geographical position data are applied in the first place. There are also solutions where maps based on 3D lidar data [114] and image semantics [115] are used.

3.4.2. Methods of planning and control of vehicle movement

According to [116], methods of planning and control of vehicle movement fall into the following categories:

- those where advanced methods of perception and decision making are utilized sequential planning, with the following classification:
    - traditional methods:
        - non-linear control [117] consisting in tracking the trajectory of the vehicle movement and maintaining it by steering the vehicle along the designated route;
        - Model Predictive Control (MPC) [118] consisting in precise determination of vehicle trajectory using traffic optimization and prediction techniques;
        - feedback-feedforward control [119], the aim of which, in feedforward control, is to estimate the steering angle required to drive on the road of known curvature and speed profile, and in feedback control, to minimize the feedforward control error;
        - shared control or parallel autonomy, in which the autonomous system functions as a *guardian angel*, providing additional safety by taking over the driver's tasks in dangerous situations. Systems of parallel autonomy may also take over vehicle operation at the driver's request, and this is referred to as interleaved autonomy [116];
    - *end-to-end* methods based on machine learning which replace the traditional algorithms of environment perception (described in sections 3.1, 3.1.1, 3.1.2, 3.1.3, 3.2, 3.2.1 and 3.3). These end-to-end methods apply to single components of the system (as distinguished from the end-to-end autonomy described further on), e.g. to keeping the vehicle within a lane [118];
- behaviour-aware planning can be treated as an open challenge due to the amount of interaction between the various traffic users. Socially compliant driving and cooperation with real human drivers is of great importance for safety in uncertain environments [116]. Situations such as turning left at a congested road, driving through a narrow passage with no traffic lights (alternating traffic), incorrect road marking and other drivers' mistakes are just examples of situations that a fully autonomous road vehicle and decision making system must cope with. This article does not describe solutions for unusual traffic situations, but there have been a number of attempts to solve and describe these problems, e.g. [121], [122], [123];
- end-to-end planning (end-to-end autonomy), which is a solution based on machine learning and on a holistic approach, i.e. without any breakdown into components responsible for some part of the system (e.g. keeping the vehicle within a lane). There are systems in which a convolutional neural network learns to carry out perception and generates the route solely on the basis of the driving sequence and the lidar data [124]. A similar solution utilizing camera images, lidar and semantic image segmentation (KITTI data model [125]) was developed the same year [126].

3.4.3. Decision making mechanisms

Driving decision-making mechanisms (DDM) are considered a key technology in ensuring the autonomous vehicle driving safety. The DDM mechanisms determine the vehicle movement strategy based on the information collected, i.e. the results of applying algorithms used in autonomous vehicle systems (mentioned above in the document) and sensor data.

The author has not aimed to accurately classify the decision-making mechanisms due to the large number of solutions applied that depend on the capability (or necessity) of using decision making mechanisms (e.g. individual decisions related to identification of road signs and reliability of estimates or a series of decisions related to vehicle travel to the point of destination), separate system components (e.g. maintaining safe distance to another vehicle, active cruise control ACC) and the level of automation. It is, however, possible to present some decision-making models that may be based on:

- decision trees [127], [128] and diagrams [129];
- Partially Observable Markov Decision Processes (POMPD) [130];
- machine learning:
  - Support Vector Machine Regression (SVR) with Particle Swarm Optimization (PSO) [131];
  - Markov Decision Processes (MPD) with Reinforcement Learning (RL) [132];
  - Deep Reinforcement Learning (DRL) [133].

3.4.4. Artificial intelligence

Traffic planning, vehicle control and decision making using machine learning methods only, i.e. end-to-end autonomy, is a very simplified approach, taking into account the number of algorithms for the detection, identification, tracking of objects, environment mapping and reacting to unusual events (discussed in sections 3.1 to 3.4.3). In theory, artificial intelligence can solve any of the problems discussed, but its effectiveness would depend on the number and selection of training data, i.e. the examples that the artificial intelligence needs to learn. Another problem, certainly a solvable one, may be the need to carry out appropriate tests to check the safety of such a system in different situations, under different conditions and in different locations. Currently, what can be learned from [134], methods of testing autonomous vehicle systems are not developed in a reliable way and neither are they very demanding.

The end-to-end autonomy was already implemented in 1989 in an ALVINN (Autonomous Land Vehicle in a Neural Network) system [135]. NVIDIA used deep machine learning (convolutional neural networks) in 2016 [136]. Nowadays imitation learning techniques [137] are becoming popular, in which the network learns on the basis of images that imitate reality (simulators) [138].

4. SUMMARY

Autonomous vehicle systems combine various techniques of perceiving the environment using popular sensors [75] and algorithms. Methods that solve specific tasks (environment recognition, planning, decision-making) use analytical algorithms as well as methods of artificial intelligence, which in some cases produce much better results. Replacing all algorithms with artificial intelligence (end-to-end autonomy) does not seem to be a viable solution today because that would require gathering huge amounts of adequate data.

It will certainly be a long time before fully autonomous vehicles populate the roads. The process will impose changes in traffic law, create issues of liability for accidents caused by autonomous vehicles, ethical issues (the trolley problem) and will require determining the rules of the driving licence test for this type of driver. Nevertheless, the technological progress in IT will have a continuous impact on the choice of algorithms used in autonomous vehicle systems and on the emergence of novel methods and algorithms.

This summary does not provide a list of the best algorithms to be used in autonomous vehicles, as the choice of such algorithms depends on:

- computational power of the equipment used, which affects the speed of algorithm execution;
- type of sensors, which depends on the dimensions and use of the vehicle;
- quality of the sensors, which is determined by the available budget (a 3D lidar with a 300 metres range costs more than ten thousand dollars);
- level of automation.

The aim of the article (based on a literature review and literature studies) was to collect information on algorithms used so far in autonomous vehicle systems, including algorithms developed for use in practical solutions. The extensive bibliography on the topic may serve as a guidepost for those interested in looking deeper into the algorithmics of systems used in autonomous vehicles.

5. REFERENCES

[1] Ziomek O.: Volkswagen będzie wypożyczał samochody elektryczne na minuty, zacznie w 2019 roku. Portal DobreProgramy, 2018. https://www.dobreprogramy.pl/Volkswagen-bedzie-wypozyczac-samochody-elektryczne-na-minuty-zacznie-w-2019-roku,News,89124.html [Retrieved: 20.09.2018].

[2] National Research Council USA: Technology Development for Army Unmanned Ground Vehicles. Washington, 2003.

[3] Kardasz P., Lyubov O. Kardasz E.: Autonomiczne pojazdy. Biuletyn naukowy Warszawskiej Wyższej Szkoły Informatyki Stosowanej. Informatyka (7). 2017. ISSN: 2082-9892.

[4] Berrada J.: Modeling Transportation Systems involving Autonomous Vehicles: A State of the Art. Elsevier, 2017.

[5] Engineering.com: The Road to Driverless Cars: 1925-2025. https://www.engineering.com/DesignerEdge/DesignerEdgeArticles/ArticleID/12665/The-Road-to-Driverless-Cars-1925--2025.aspx [Retrieved: 28.08.2018].

[6] Felton R.: The Man Who Tested The First Driverless Car in 1925 Had A Bizarre Feud With Harry Houdini. https://jalopnik.com/the-man-who-tested-the-first-driverless-car-in-1925-had-1792312207 [Retrieved: 31.08.2018].

[7] GM Heritage Center: Self-Driving Cars, in 1956? https://www.gmheritagecenter.com/featured/Autonomous_Vehicles.html [Retrieved: 28.08.2018].

[8] CyberneticZoo.com: 1960 – Stanford Cart. http://cyberneticzoo.com/cyberneticanimals/1960-stanford-cart-american/ [Retrieved: 28.08.2018].

[9] Gupta N., S. N. Dana: On the complexity of blocks-world planning. Artificial Intelligence (56). Elsevier, 1992. http://www.cs.umd.edu/~nau/papers/gupta1992complexity.pdf [Retrieved: 28.08.2018].

[10] Gupta A., Efros A., M. Hebert: Blocks World Revisited: Understanding Using Qualitive Geometry and Mechanics. European Conference on Computer Vision. 2010. http://www.cs.cmu.edu/~abhinavg/blocksworld/blocksworld.pdf [Retrieved: 28.08.2018].

[11] Kuipers B. J.: An Hyphothesis-Driven Recognition System for the Blocks World. MIT, 1974.

[12] Dickmanns E. D.: Vehicles Capable of Dynamic Vision. Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence. Nagoya, 1997.

[13] Dickmanns E. D.: Expectation-Based, Multi-Focal, Saccadic (EMS) Vision for Dynamic Scene Understanding. Control Engineering Practise (10), nr 8. Elsevier, 2012.

[14] Gregor R., Lutzeler M., Pellkofer M. Siedersberg K.-H, Dickmanns E. D.: EMS-Vision: A Receptual System for Autonomous Vehicles. Transactions on Intelligent Transportation Systems (3), nr 1. IEEE, 2002.

[15] Hofmann U., Rieder A., Dickmanns E. D.: EMS-Vision: Application to Hybrid Adaptive Cruise Control. Proceedings of the IEEE Inteligent Vehicles Symposium 2000. IEEE, 2000. ISBN: 0-7803-6363-9.

[16] Launchbury J.: A Darpa Perspective on Artificial Intelligence. 2017. https://www.youtube.com/watch?v=-O01G3tSYpU [Retrieved: 29.08.2018].

[17] Markoff J.: Google Cars Drive Themselves, in Traffic. 2010. https://www.nytimes.com/2010/10/10/science/10google.html [Retrieved: 29.08.2018].

[18] Russel S.: DARPA Grand Challenge Winner, Popular Mechanics. 2006. https://www.popularmechanics.com/technology/robots/a393/2169012/ [Retrieved: 29.08.2018].

[19] Cargegie Mellon University: Carnegie Mellon's Tartan Racing Wins $2M DARPA Urban Challenge. https://www.cmu.edu/homepage/practical/2007/fall/victory.shtml [Retrieved: 30.08.2018].

[20] Cargegie Mellon University: Tartan Racing Technology. http://www.cs.cmu.edu/~tartanrace/tech.html [Retrieved: 30.08.2018].

[21] Tesla Motors: Enhancing Safety and Convenience with Summon. 2016. https://www.tesla.com/blog/enhancing-safety-and-convenience-summon [Retrieved: 30.08.2018].

[22] NHTS: Preliminary Statement of Policy Concerning Automated Vehicles. 2017. https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety [Retrieved: 28.08.2018].

[23] Zhang B.: ELON MUSK: In 2 years your Tesla will be able to drive from New York to LA and find you. 2016. https://finance.yahoo.com/news/elon-musk-two-years-car-202858960.html [Retrieved: 30.08.2018].

[24] Cunningham W.: 2013 Ford Fusion review. 2013. https://www.cnet.com/roadshow/reviews/2013-ford-fusion-review/ [Retrieved: 30.08.2018].

[25] English A.: New Car Tech: 2014 Mercedes-Benz S-Class. 2013. https://www.roadandtrack.com/new-cars/news/a3689/new-car-tech-2014-mercedes-benz-s-class/.

[26] Tregaskis S.: The first generation of self-driving cars – in pictures. 2014. https://www.theguardian.com/technology/gallery/2014/may/29/first-generation-self-driving-cars-google-in-pictures.

[27] Hirz M., Walzel B.: Sensor and object recognition technologies for self-driving cars. Computer-Aided Design and Application. 2018.

[28] Rasmussen C.: Grouping dominant orientations or Ill-structured road following. International Conference on Computer Vision and Pattern Recognition. IEEE, 2004.

[29] Miksik O.: Rapid Vanishing Point Estimation for General Road Detection. Proceedings – IEEE International Conference on Robotics and Automation. IEEE, 2012.

[30] Moon Y. Y., Geem Z. W., Han G. T.: Vanishing point detection for self-driving car using harmony search algorithms. Swarm and Evolutionary Computation. 2018.

[31] Pohlen T., Hermans A., Mathias M., Leibe B.: Full Resolution Residual Networks. 2017. http://openaccess.thecvf.com/content_cvpr_2017/papers/Pohlen_Full-Resolution_Residual_Networks_CVPR_2017_paper.pdf [Retrieved: 3.09.2018].

[32] Petrovskaya A., Thrun S.: Model based vehicle detection and tracking for autonomous urban driving. Autonomous Robots (26). Springer, 2009.

[33] Trempl M., Arjona-Medina J., Unterthiner T.: Speeding up Semantic Segmentation for Autonomous Driving. 2016. https://openreview.net/pdf?id=S1uHiFyyg [Retrieved: 3.09.2018].

[34] Mukhometzianov R., Wang Y.: Review. Machine learning techniques for traffic sign detection. 2017. https://arxiv.org/ftp/arxiv/papers/1712/1712.04391.pdf [Retrieved: 4.09.2018].

[35] Nvidia: Nvidia Drive. https://www.nvidia.com/en-us/self-driving-cars/drive-platform/ [Retrieved: 3.09.2018].

[36] Hruska J.: Tesla Working with AMD on Self-Driving Car Processor. https://www.extremetech.com/extreme/256171-tesla-working-amd-self-driving-car [Retrieved: 3.09.2018].

[37] Zhang H., Qi X., Shen X., Jia J.: ICNet for Real-Time Semantic Segmentation on High-Resolution Images. 2018. https://arxiv.org/pdf/1704.08545.pdf [Retrieved: 3.09.2018].

[38] K. Brkić: An overview of traffic sign detection methods. https://pdfs.semanticscholar.org/74a1/336f1fbc8b7bb3b6e159711af1a91336ce22.pdf [Retrieved: 4.09.2018].

[39] T. Radu, Z. Karel, L. V. Gool: Multi-view traffic sign detection, recognition, and 3D localisation. 2009 Workshop on Applications of Computer Vision (WACV). 2009.

[40] Sugiharto A., Harjoko A.: Traffic sign detection based on HOG and PHOG using binary SVM and k-NN. 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE). 2016.

[41] Fleyeh H., Biswas R., Davami E.: Traffic sign detection based on AdaBoost color segmentation and SVM classification. IEEE, 2013.

[42] Gurel C. S.: Traffic Sign Recognition using multi-class SVM. Perception for Autonomous Robots. 2018.

[43] Samuele S., Alioscia P., Federico T., Nicola F., Luigi S.: A traffic sign detection pipeline based on interest region extraction. The 2013 International Joint Conference on Neural Networks (IJCNN). 2013.

[44] Martinez A. Kak A., PCA versus LDA. IEEE Trans. Pattern Analysis and Machine Intelligence (23), nr 2. IEEE, 2001.

[45] Wang G., Ren G., Wu Z., Zhao Y., JiangL.: A robust, coarse-to-fine traffic sign detection method. The 2013 International Joint Conference on Neural Networks (IJCNN). 2013.

[46] Shi J., Lin H.-Y.: A vision system for traffic sign detection and recognition. 26th International Symposium on Industrial Electronics (ISIE). IEEE, 2017.

[47] Fukazawa Y.: Traffic Sign Recognition system on Android devices. 2003. https://www.researchgate.net/publication/311540973_Traffic_Sign_Recognition_system_on_Android_devices [Retrieved: 4.09.2018].

[48] Jagannathan S., Desappan K., Swami P.: Efficient Object Detection and Classification on Low Power Embedded Systems. International Conference on Consumer Electronics (ICCE). IEEE, 2017.

[49] Li Z., Dong C., Zheng L., Liu L.: Traffic Signs Detection Based on Haar-Like Features and Adaboost Classifier. Second International Conference on Transportation Information and Safety. 2013.

[50] Kaplan K., Kurtul C., Akin H. L.:Real-Time Traffic Sign Detection and Classification Method for Intelligent Vehicles. International Conference on Vehicular Electronics and Safety. IEEE, 2012.

[51] Nguyen K.-D., Le D.-D, Duc D. A.: Efficient Traffic Sign Detection Using Bag of Visual Words and Multi-scales SIFT. International Conference on Neural Information Processing, 2013.

[52] De Souza A., Fontana C., Mutz F.: Traffic sign detection with VG-RAM weightless neural networks. The 2013 International Joint Conference on Neural Networks (IJCNN). 2013.

[53] Zhang K., Sheng Y., Wang P.: Automatic recognition of traffic signs in natural scene image based on central projection transformation. http://www.isprs.org/proceedings/XXXVII/congress/3b_pdf/26.pdf [Retrieved: 4.09.2018].

[54] Shustanov A., Yakimov P.: CNN Design for Real-Time Traffic Sign Recognition. Procedia Engineering (201). ELSEVIER, 2017.

[55] Wu Y., Liu Y., Li J., Liu H., Hu X.: Traffic sign detection based on convolutional neural networks. The 2013 International Joint Conference on Neural Networks (IJCNN). 2013.

[56] Zhu Z., Liang D., Zhang S.: Traffic-Sign Detection and Classification in the Wild. https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Zhu_Traffic-Sign_Detection_and_CVPR_2016_paper.pdf [Retrieved: 4.09.2018].

[57] Bappy J. H., Roy-Chowdhury K.: CNN based region proposals for efficient object detection. http://intra.ece.ucr.edu/~mbappy/pubs/ICIP_2016.pdf [Retrieved: 4.09.2018].

[58] Xiong C., Wang C., Ma W., Shan Y.: A traffic sign detection algorithm based on deep convolutional neural network. International Conference on Signal and Image Processing (ICSIP). IEEE, 2016.

[59] Zuo Z., Yu K., Zhou Q., Wang X., Li T.: Traffic Signs Detection Based on Faster R-CNN. 37th International Conference on Distributed Computing Systems Workshops (ICDCSW). 2017.

[60] Durrant-Whyte H., Bailey T.: Simultaneous localisation and mapping (slam): Part I The essential algorithms. Robotics and Automation Magazine. 2006.

[61] Smith R., Cheeseman P.: On the representation and estimation of spatial uncertainty. Int. Journal of Robotics Research. 1987.

[62] Montemerlo M., Thrun S., Koller D., Wegbreit B.: FastSLAM: A factored solution to the simultaneous localization and mapping problem. AAAI National Conference on Artificial Intelligence, 2002.

[63] Doucet A., De Freitas N., Murphy K., Russel S.: Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. Uncertainty in Artificial Intelligence Proceedings. 2000.

[64] Montemerlo M., Thrun S.: FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges. Proceedings of IJCAI. 2003.

[65] Ebdres F., Hess J., Engelhard N.: An evaluation of the RGB-D SLAM system. International Conference on Robotics and Automation. IEEE, 2012.

[66] Mur-Artal E., Tardos J.D.: ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. Transactions on Robotics PP (99). IEEE, 2016.

[67] Rublee E., Rabaud V., Konolige K., Bradski G.: ORB: an efficient alternative to SIFT or SURF. http://www.willowgarage.com/sites/default/files/orb_final.pdf, [Retrieved: 5.09.2018].

[68] Ball D., Heath S., Wiles J.: OpenRatSLAM: an open source brain-based SLAM system. Automonous Robots. 2013.

[69] Engel J., Shöps T., Cremers D.: LSD-SLAM: Large-Scale Direct Monocular SLAM. Computer Vision. ECCV, 2014.

[70] Petridis V., Zikos N.: L-SLAM: „Reduced dimensionality FastSLAM algorithms. The 2010 International Joint Conference on Neutral Networks (IJCNN). 2010.

[71] Yang S., Shrerer S. A., Zell A.: Visual SLAM for autonomous MAVs with dual cameras. International Conference on Robotics and Automation. IEEE, 2014.

[72] Levinson J., Askeland J., Becker J.: Towards Fully Autonomous Driving: Systems and Algorithms. Intelligent Vehicles Symposium (IV). IEEE, 2011.

[73] Teichman A., Thrun S.: Practical object recognition in autonomous driving and beyond. Advanced Robotics and its Social Impacts. IEEE, 2012.

[74] Ors A. O.: Radar, camera, LiDAR and V2X for autonomous cars. 2017. https://blog.nxp.com/automotive/radar-camera-and-lidar-for-autonomous-cars [Retrieved: 03.09.2018].

[75] Rainer M.: Indoor positioning technologies. ETH Zürich, 2012. https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/54888/eth-5659-01.pdf [Retrieved: 27-08-2018]

[76] Thrun S., Montemerlo M.: The GraphSLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures. The International Journal of Robotics Research. 2006.

[77] Colleens T., Colleens J. J., Ryan C.: Occupancy grid mapping: An empirical evaluation. Control & Automation. IEEE, 2007.

[78] Shuster F., Keller C., Rapp M. Haueis M.: Curio C.: Landmark based radar slam using graph optimization. Intelligent Transportation Systems (ITSC). IEEE, 2016.

[79] Schoen M., Horn M., Hahn M., Dickmann J.: Real-Time Radar SLAM. FAS, 2017. https://www.uni-das.de/images/pdf/veroeffentlichungen/2017/01.pdf [Retrieved: 6.09.2018].

[80] Thrun S., Burgard W., Fox D.: A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. International Conference on Robotics and Automation. IEEE, 2000.

[81] Smets P., Kennes R.: The transferable belief model. Artificial Inteligence (66), No. 2. Elsevier, 1994.

[82] Trehard C., Pollard E., Bradai B., Nashashibi F.: Credibilist SLAM Performances with Different Laser Set-ups. 13th Interational Conference on Control, Automation, Robotics and Vision (ICARCV). 2014.

[83] Besl P. J., McKay H. D.: A method for registration of 3-D shapes,. IEEE PAMI (14), nr 2. IEEE, 1992.

[84] Mendes E., Koch P., Lacroix S.: ICP-based pose-graph SLAM. International Symposium on Safety, Security, and Rescue Robotics (SSRR). IEEE, 2016.

[85] Datta A.: Google releases SLAM tool Cartographer to open source community. Geospatial World, 2016. https://www.geospatialworld.net/blogs/google-open-sources-slam-tool-cartographer/ [Retrieved: 10.09.2018].

[86] Nuchter A., Bleier M., Schauer J., Janotta P.: Impoving Google's Cartographer 3D Mapping by Continuous-Time SLAM. https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W3/543/2017/isprs-archives-XLII-2-W3-543-2017.pdf [Retrieved: 10.09.2018].

[87] Zhang J., Singh S.: LOAM: Lidar Odometry and Mapping in Real-time. Conference: Robotics: Science and Systems Conference. 2014.

[88] Rodriguez-Canosa R. G., Thomas S., del Cerro J.: A Real-Time Method to Detect and Track Moving Objects (DATMO) from Unmanned Aerial Vehicles (UAVs) Using a Single Camera. Remote Sensing. 2012.

[89] Vivet D., Checchin P., Chapuis R.: A Mobile Ground-based Radar Sensor of Detection And Tracking of Moving Objects. EURASIP Journal on Advances in Signal Processing 2012 (1). 2012.

[90] Magnier V., Gruyer D., Godelle J.: Automotive LIDAR objects Detection and Classification Algorithm Using the Belief Theory. Intelligent Vehicles Symposium. IEEE, 2017.

[91] Vu T.-D.:Vehicle perception: Localization, mapping with detection, classification and tracking of moving objects. Ph.D. dissertation. Institut National Polytechnique de Grenoble (INPG), 2009.

[92] Pancham A., Tlale N., Bright G.: Literature Review of SLAM and DATMO. 2011. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.886.7941&rep=rep1&type=pdf [Retrieved: 10.09.2018].

[93] Wang C.-C., Thorpe C., Thrun S.: Online Simultaneous Localization and Mapping with Detection and Tracking of Moving Objects: Theory and Results from a Ground Vehicle in Crowded Urban Areas. International Conference on Robotics and Automation (ICRA). IEEE, 2003.

[94] Kim C., Li F., Ciptadi A., Rehg J. M.: Multiple Hypothesis Tracing Revisited. International Conference on Computer Vision (ICCV). IEEE, 2015.

[95] Watson G. A., Blair W. D.: IMM Algorithm for Tracking Targets that Maneuver through Coordinated Turns. Proceedings of SPIE. 1992.

[96] Hähnel D., Schulz D., Burgard W.: Mobile robot mapping in populated environments. http://ais.informatik.uni-freiburg.de/publications/papers/haehnel-populated.pdf [Retrieved: 11.09.2018].

[97] Rezatofighi S. H., Milan A., Zhang Z.: Joint Probabilistic Data Association Revisited. https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Rezatofighi_Joint_Probabilistic_Data_ICCV_2015_paper.pdf [Retrieved: 11.09.2018].

[98] Montesano L., Minguez J., Montano L.: Modeling the static and the dynamic parts of the environment to improve sensor-based navigation. International Conference on Robotics and Automation (ICRA). IEEE, 2005.

[99] Cover T. M.: Estimation by the Nearest Neighbor Rule. 1968. https://pdfs.semanticscholar.org/32ba/0664f51c88960fc9b783a9c0adb6b0546e73.pdf [Retrieved: 11.09.2018].

[100] Solà J.: Towards Visual Localization, Mapping and Moving Objects Tracking by a Mobile Robot: a Geometric and Probabilistic Approach. 2007. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.125.7858&rep=rep1&type=pdf [Retrieved: 11.09.2018].

[101] Vu T. -D., Burlet J., Aycard O.: Grid-based localization and local mapping with moving object detection and tracking. Information Fusion 12, 2011.

[102] Na K., Byun J., Roh M., Seo B.: RoadPlot-DATMO: Moving object tracking and track fusion system using multiple sensors. 2015 International Conference on Connected Vehicles and Expo (ICCVE), 2015.

[103] Botha F. J.: Data Fusion of Radar and Stereo Vision for Detection and Tracking of Moving Objects. PRASA-RobMech International Conference. 2016.

[104] Holstein T., Crnkovic G. D., Pelliccione P.: Ethical and Social Aspects of Self-Driving Cars. ARXIV. 2018.

[105] Holstein T.: The Misconception of Ethical Dilemmas in Self-Driving Cars. Proceedings. 2017.

[106] Goodall H. J.: Ethical Decision Making During Automated Vehicle Crashes. Transportation Research Record Journal of the Transportation Research Board. 2014.

[107] Bellman R.: On a routing problem. 1956. http://www.dtic.mil/dtic/tr/fulltext/u2/606258.pdf [Retrieved: 19.09.2018].

[108] Ford L. R.: Network Flow Theory. 1958. http://www.dtic.mil/dtic/tr/fulltext/u2/422842.pdf [Retrieved: 19.09.2018]

[109] Dijkstra E. W.: A note on two problems in connexion with graphs. 1959. http://www-m3.ma.tum.de/foswiki/pub/MN0506/WebHome/dijkstra.pdf [Retrieved: 19.09.2018]

[110] Hart P. E., Nilsson N. J., Raphael B.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths. Transactions on Systems Science and Cybernetics (4). IEEE, 1968.

[111] Stentz A., Mellon I. C.: Optimal and efficient path planning for unknown and dynamic environments. International Journal of Robotics and Automation. 1993.

[112] Dolgov D., Thrun S., Montemerlo M.: Practical Search Techniques in Path Planning for Autonomous Driving. 2008. https://ai.stanford.edu/~ddolgov/papers/dolgov_gpp_stair08.pdf [Retrieved: 19.09.2018].

[113] Chu K., Kim j., Jo K., Sunwoo M.: Real-time path planning of autonomous vehicles for unstructured road navigation. International Journal of Automotive Technology. 2015.

[114] Wang L., Zhang Y, Wang J.: Map-Based Localization Method for Autonomous Vehicles Using 3D-LIDAR. The 25th International Symposium on Dynamics of Vehicles on Roads and Tracks (IAVSD). 2017.

[115] Xiao Z., Jiang K., Xie S.: Monocular Vehicle Self-localization method based on Compact Semantic Map. 2018.

[116] Schwarting W., Mora J. A., Rus D.: Planning and Decision-Making for Autonomous Vehicles. Annual Reviews. 2018.

[117] Hoffmann G., Tomlin C. J., Montemerlo M., Thrun S.: Autonomous Automobile Trajectory Tracking for Off-Road Driving: Controller Design, Experimental Validation and Racing. Proceedings of the American Control Conference. IEEE, 2007.

[118] Keviczky T., Falcone O., Borrelli F.: Predictive Control Approach to Autonomous Vehicle Steering. Transactions on Control Systems Technology. IEEE, 2007.

[119] Kapania N. R., Gerdes J. C.: Design of a feedback-feedforward steering controller for accurate path tracking and stability at the limits of handling. 2015. https://www.tandfonline.com/doi/abs/10.1080/00423114.2015.1055279 [Retrieved: 11.09.2018].

[120] Sallab A. E., Abdou M., Perot E., Yogamani S.: End-to-End Deep Reinforcement Learning for Lane Keeping Assist. 2016. https://openreview.net/pdf?id=ByjDCQgke [Retrieved: 12.09.2018].

[121] Aoki S., Rajkumar E.: Dynamic Intersections and Self-Driving Vehicles. 2018. https://www.researchgate.net/publication/326145557_Dynamic_Intersections_and_Self-Driving_Vehicles [Retrieved: 12.09.2018].

[122] Krishnan S., Govind A. R., Ramakrishnan R.: A Look at Motion Planning for Avs at an Intersection. 2018. https://arxiv.org/pdf/1806.07834.pdf [Retrieved: 12.09.2018].

[123] Isele D., Rahimi R., Cosgun A.: Navigation Occluded Intersections with Autonomous Vehicles using Deep Reinforcement Learning. 2018. https://arxiv.org/pdf/1705.01196.pdf [Retrieved: 12.09.2018].

[124] Caltagirone L., Bellone M., Svensson M.: LIDAR-based driving path generation using fully convolutional neural networks. 2017. https://www.researchgate.net/publication/315666984_LIDAR-based_Driving_Path_Generation_Using_Fully_Convolutional_Neural_Networks [Retrieved: 12.09.2018].

[125] The KITTI Vision Benchmark Suite: Semantic and Instance Segmentation Evaluation. http://www.cvlibs.net/datasets/kitti/ [Retrieved: 12.09.2018].

[126] Barnes D., Maddern W., Posner W.: Find your own way: weakly-supervised segmentation of path proposals for urban autonomy. International Conference on Robotics and Automation (ICRA). IEEE, 2017.

[127] Olsson M.: Behavior Trees for decision-making in Autonomous Driving. 2016. http://www.diva-portal.org/smash/get/diva2:907048/FULLTEXT01.pdf [Retrieved: 13.09.2018].

[128] Hu M., Liao Y., Wang W.: Decision Tree-Based Maneuver Prediction for Driver Rear-End Risk-Avoidance Behaviors in Cut-In Scenarios. Journal of Advanced Transportation. 2017.

[129] Claussmann L., Carvalho A., Schildbach G.: A path planner for autonomous driving on highways using a human mimicry approach with binary decision diagrams. European Control Conference. 2015.

[130] Ulbrich S., Mauerer M.: Probabilistic Online POMDP Decision Making for Lane Changes in Fully Automated Driving. 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). IEEE, 2013.

[131] Zhang J., Liao Y., Wang S., Han J.: Study on Driving Decision-Making Mechanism of Autonomous Vehicle Based on an Optimized Support Vector Machine Regression. Applied Sciences. 2017.

[132] Mueller M. A.: Reinforcement Learning: MDP Applied to Autonomous Navigation. 2017. http://aircconline.com/mlaij/V4N4/4417mlaij01.pdf [Retrieved: 13.09.2018].

[133] Legrand N.: Deep Reinforcement Learning for Autonomous Vehicle Control among Human Drivers. 2017. https://ai.vub.ac.be/sites/default/files/thesis_legrand.pdf [Retrieved: 13.09.2018].

[134] Harris M.: How Google's Autonomous Car Passed the First U.S. State Self-Driveing Test. IEEE Spectrum. IEEE, 2014. https://spectrum.ieee.org/transportation/advanced-cars/how-googles-autonomous-car-passed-the-first-us-state-selfdriving-test [Retrieved: 12.09.2018].

[135] Pomerleau D. A.: ALVINN: An Autonomous Land Vehicle in a Neural Network. 1989. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.830.2188&rep=rep1&type=pdf [Retrieved: 12.09.2018].

[136] Bojarski M., Del Testa D., Dworakowski D.: End to end learning for self-driving cars. https://arxiv.org/abs/1604.07316 [Retrieved: 12.09.2018].

[137] Hussein A., Gaber M. M., Elyan E.: Imitation Learning: A Survey of Learning Methods. ACM Computer Surveys. 2017.

[138] Dosovitskiy A, Ros G., Codevilla F.: CARLA: An Open Urban Driving Simulator. 2017. http://proceedings.mlr.press/v78/dosovitskiy17a/dosovitskiy17a.pdf [Retrieved: 12.09.2018].