Autonomous Driving Architectures, Perception and Data Fusion: A Review

Conference Paper · September 2020

DOI: 10.1109/ICCP51029.2020.9266268

4 authors:

Gustavo Velasco-Hernandez
Institute of Technology, Tralee

De Jong Yeong
Munster Technological University

John Barry
Liebherr-International Deutschland GmbH

Joseph Walsh
Munster Technological University

Autonomous Driving Architectures, Perception and Data Fusion: A Review

Gustavo Velasco-Hernandez*†, De Jong Yeong*, John Barry* and Joseph Walsh*
*IMaR Research Centre / Lero, The Irish Software Research Centre
Institute of Technology Tralee
Tralee, Ireland
†gustavo.velascohernandez@staff.ittralee.ie
ORCID 0000-0002-2177-6348

***Abstract*—Over the last 10 years, huge advances have been made in the areas of sensor technologies and processing platforms, pushing forward developments in the field of autonomous vehicles, mostly represented by self-driving cars. However, the complexity of these systems has been also increased in terms of the hardware and software within them, especially for the perception stage in which the goal is to create a reliable representation of the vehicle and the world. In order to manage this complexity, several architectural models have been proposed as guidelines to design, develop, operate and deploy self-driving solutions for real applications. In this work, a review on autonomous driving architectures is presented, classifying them into technical or functional architectures depending on the domain of their definition. In addition, the perception stage of self-driving solutions is analysed as a component of the architectures, detailing into the sensing part and how data fusion is used to perform localisation, mapping and object detection. Finally, the paper is concluded with additional thoughts on the actual status and future trends in the field.**

***Index Terms*—Architecture, Autonomous Driving, Autonomous Vehicles, Data Fusion, Localisation, Mapping, Perception, Self-Driving Car, Sensor Fusion.**

I. Introduction

Several initiatives for the development of self-driving solutions have been created in industry and academia, but there is not a one-size-fits-all approach that could accomplish all the goals or to solve all the issues present in each application scenario. However, there are certain components and processing stages that are shared among different projects in order to complete the main objective of such a system, this is, autonomous navigation of a platform in a specified environment.

On the component side, there are hardware parts, like positioning and range sensors, networks, embedded and High-Performance Computing (HPC) platforms, and also software components, low-level embedded and high-level application software. Regarding the processing stages, they can be summarised in these four categories: sensing and perception, processing and planning, vehicle control, and system supervision. No matter whether the platform is a car on the road, a robot in a warehouse, a tractor in a crop field or a lifting vehicle in a building site, these stages are part of any self-driving architecture definition. The differences between them will be the conditions where they will operate, encompassed in their Operational Design Domain (ODD), which defines the scope and limitations of the environment where a self-driving system or feature should work, including but not limited to weather, terrain, time-of-day, location, etc.

An overall description of the common elements of a self-driving solution can be found in [1] and [2], however, it is required to organise these elements in a way that enables a successful product development life cycle, and avoid reinventing the wheel for every self-driving development to be done. For this reason, several authors have proposed architectural models for the design, development, and deployment of autonomous driving systems, both from a technical and a functional point of view.

Some of these initiatives have a wider scope while others focus on certain specific aspects. One of these aspects is the perception stage of autonomous vehicles, where multiple sensors provide information using different modalities with the objective of creating a reliable representation of the vehicle, its status and the world that surrounds it. This supposes a challenge because all the sensors could be seen as separate subsystems generating streams of data, but this data needs to be merged in an appropriate way. This process is known as data fusion and it handles the collection, association of data and the creation of better representations than those obtained by using the sensors individually.

The rest of this paper is structured as follows: Section II provides an overview of autonomous driving architectures found in the literature. These are classified based on their domain or abstraction level (functional vs technical) and each one of their inner elements is described. Additional concepts on sensors, perception and data fusion are presented in section III, starting with a description of typical sensors used in a self-driving environment and then, different developments in the field of perception are highlighted, especially in two areas: localisation and mapping, and object detection. Then an overview of data fusion, its relevance and its challenges is presented, including recent work in the area. Finally, conclusions are presented in section IV with additional thoughts on the actual status and future trends on autonomous vehicles, especially in perception and data fusion as part of a self-driving architecture.

II. Autonomous Driving Architectures

When representing a complex system using an architectural model, there are different perspectives from which the system can be viewed, for example in terms of physical components, development stages, logical functions or process blocks, as described in [3]. In the present work, autonomous driving architectures are considered from two viewpoints: 1) a technical viewpoint, which is related to hardware and software components and their implementations, and 2) from a functional viewpoint, which includes the description of processing stages that a self-driving vehicle must have as logical blocks of the whole system.

A. Technical View

Hardware and software are the two main layers of the technical view of an autonomous driving architecture and each layer includes components that represent different aspects of the whole system. Some of these components can be seen as an isolated group, but there are some components that act like a backbone within their own layer providing a structure and guidelines for the interactions between all the components. This description is depicted in figure 1.

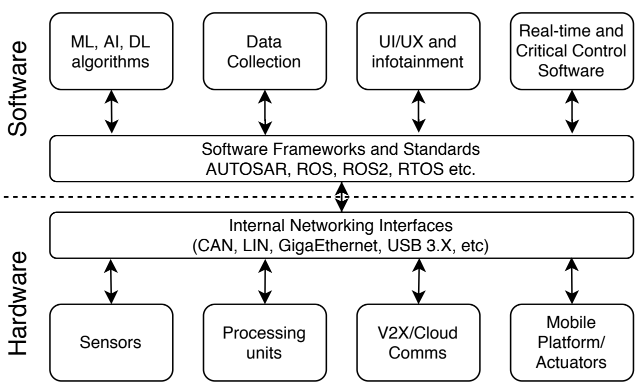

Fig. 1. Technical architecture for an autonomous driving system

Autonomous vehicles nowadays are large complex systems equipped with several *sensors*, for internal and external monitoring and generating massive amounts of data per day. In order to handle all that information, communications and *processing units* in the vehicles are no longer limited to a number of Electronic Control Units (ECUs) with low-bandwidth networks as it used to be. More powerful devices are used to collect and process the data coming from the sensors, like heterogeneous computing platforms with multiple cores, Graphical Processing Units (GPUs) and Field Programmable Gate Arrays (FPGAs). In addition to the data generated by the vehicle, external, data is also available from the internet, other vehicles or infrastructure, what is known as *Vehicle-to-Anything communications (V2X)*. The hardware part also includes the vehicle itself, this is the *mobile platform and actuators*, which can be of different kinds depending on the application and terrain where the system will operate. The *internal networking interfaces* allow each subsystem to exchange information with each other, for example high bandwidth interfaces like USB 3.x or Gigabit Ethernet for sensor data transport, or CAN and LIN networks for low bandwith communication.

The processing capabilities of current self-driving vehicles is such that they are sometimes they are referred to as supercomputers on wheels. This statement is not far from reality because due to the complexity in the hardware side, the software side of vehicles has been also evolving, from the embedded software running on top of Real-Time Operating Systems (RTOS) on ECUs, to high-level software including frameworks, libraries and modules that support *Machine Learning (ML), Artificial Intelligence (AI) and Deep Learning (DL) algorithms* required for processing the data. But also, there are software components dealing with different aspects of the vehicle operation, like drivers for *data collection* from the sensors, user interfacing through the *infotainment system*, and *real-time and critical software* for controlling actuators and monitoring the status of the vehicle. This complexity in software creates the need to follow certain patterns and implement standards that enable a successful development, management and deployment of such systems, both at a low-level (hard real-time software/firmware) and at a high-level (detection, inference or forecasting software). These *software frameworks and standards* provide a structured way for the software to operate in a concurrent and cooperative way.

One example of software guidelines and frameworks is the AUTomotive Open System ARchitecture, AUTOSAR [4] [5], widely used in the automotive industry. Its main goal is to create and establish an open and standardized architecture for ECUs, using a component-based software design approach. Another example is the Robot Operating System, ROS [6], a well-established software framework providing tools and libraries for robotic applications, including ready-to-use implementations for perception, navigation and motion control algorithms. However, as the robotics and self-driving landscape has changed considerably since its introduction in 2009, a new version, ROS2 [7], was re-designed based on new considerations in order to make it suitable for its use in a new range of applications, like deterministic, real-time and safety-critical systems [8]. In the case of software components as part of an autonomous driving architecture, Autoware Foundation offers its projects autoware.ai and autoware.auto [9]. They are built on top of ROS and ROS2 respectively and offer software modules for algorithms and tasks commonly used in self-driving systems.

B. Functional View

From another perspective, autonomous vehicles are composed of logical or functional blocks, which are defined based on the flow of information and the processing stages performed from data collection to the control of the vehicle, and including the internal monitoring of the system. From this, four main functional blocks can be identified across most of the proposed architectures and solutions from literature in both academia and industry: perception, planning and decision, motion and vehicle control, and system supervision. These blocks are represented in figure 2.

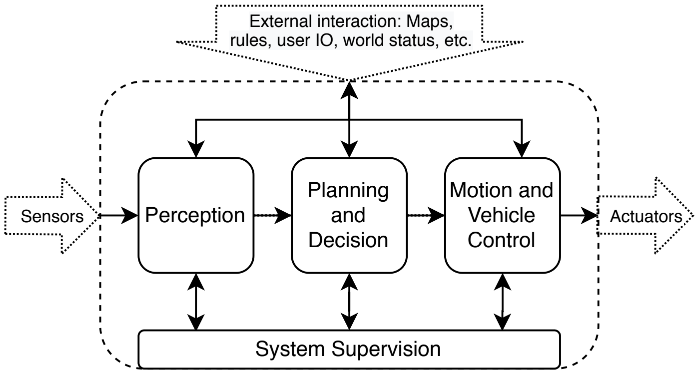

Fig. 2. Functional architecture for an autonomous driving system

The main goal of the *perception* stage is to receive data from sensors and other sources (vehicle sensors configuration, maps databases, etc), and generate a representation of the vehicle status and a world model. For performing these two tasks, sensors are categorized into proprioceptive and exteroceptive sensors. Proprioceptive sensors are those used for sensing the vehicle state, like Global Navigation Satellite Systems (GNSS), Inertial Measurement Units (IMUs), Inertial Navigation Systems (INS), and Encoders. These are used to get position, movement and odometry information of the platform. Exteroceptive sensors monitor the environment surrounding the vehicles to obtain data of the terrain, the environment and external objects. Cameras, lidars (LIght Detection And Ranging), radar and ultrasonic sensors belong to this category. After collecting all the incoming data from the sensors, two main functions are performed in the perception stage: Localisation and mapping, and object detection. More details on the perception stage are covered in section III.

Once the vehicle and world status are available for the *planning and decision* stage, the system is able to receive external information like a goal or a travel mission, and then can start the navigation plan for the vehicle, including a long-term plan (going from place A to place B in a global map or journey plan) as well as a short-term plan (execute partial goals or waypoints considering a dynamic and local map). Reactive behaviours are also included in this stage, mostly intended for the safe operation of the vehicle. Autonomous Emergency Braking (AEB) or collision preventing features are common examples of these behaviours that will override high-level commands. As most of the vehicles will interact with other actors in the road, this stage should also include external information from external sources to operate safely, information like traffic rules, maps updates and speed limits must be included to generate the plans.

The stage of *Motion and Vehicle Control* is related to the way in which the trajectory generated in the previous stage is executed on the platform, taking into account the configuration, geometry and limitations of it. The movement commands can be either goal points if the platform abstracts its configuration and the control of actuators, or movement commands like longitudinal speed, steering and braking. Again, this stage is highly associated with safety features as it receives high priority commands from the reactive behaviour modules to modify or stop the movement of the vehicle.

Another stage within an autonomous driving architecture is the *System supervision* which is in charge of monitoring all the aspects of the vehicle like the hardware (sensors operation, missing or degraded information, platform stability, energy management, fault and diagnosis, etc) and the software (autonomous driving software stack, data values within ranges, data update rates, data inconsistencies, etc). The importance of these tasks is that, as a safety-critical system, malfunctioning of hardware or software in a self-driving vehicle, should not result in any harm to people, environment or property. The ISO 26262 [10] is a standard for functional safety in road vehicles, adapted from the standard IEC 61508 [11], that aims to address possible hazards arising from faults in electric or electronic devices in a car, and offers a framework for the full product development life cycle, from specification and design to validation and production release. A further discussion on functional safety of automotive software can be found in [12].

C. Related Work

Different description of architectures can be found in literature both from a technical or a functional point of view, but also with different scopes, for example, focused on methodologies, design and concept, or focused to actual implementation on a platform. For example, reference [13] defines three different technical architectures references for automotive systems focusing on the distribution of processing and communication hardware components within a vehicle. On the software side, reference [14] presents an approach for interconnecting two environments, ROS2 and Adaptive AUTOSAR [15], frameworks that are based on a Data Distribution Service (DDS) middleware and can cooperate in the whole self-driving system.

Related also to technical architectures, there are works detailing the actual implementation of self-driving systems, both from industry and academia, like [16], which presents a brief description of a software-system architecture from the industry perspective, and in [17], the design and development of a research platform for outdoor environments is presented, which main focus is on the sensors and vehicle platform components.

On the functional side, a functional reference architecture is proposed by [18] and [19], describing three main components (perception, decision and control and platform manipulation) and two basic layers (cognitive driving intelligence and vehicle platform) within it. Another architecture is presented in [20] and its main design goal is to be compatible with safety-critical applications and standards like ISO-26262. To this end, a V-shaped, modular architecture with five layers of abstraction, providing scalability and traceability is proposed. Also, in [21] a whole framework is proposed for the design of supervision functions for the components of an autonomous driving system. The focus is on the methodological aspects of the engineering process, starting from functional and safety requirements, and generating a formal behaviour model for the specified system.

Finally, there are other works with a wider scope, addressing the technical and functional view of self-driving systems, presenting full deployments [22] and comparing different architectures [23], [24]. An example of a deployed system is found in [22] where a self-driving car architecture solution based on ROS2 is presented, addressing some limitations of the previous version of ROS in terms of usability and maintainability. On the side of architectures comparison, reference [23] presents an overview of four architectures of self-driving solutions, from an industrial case to a resource-constrained prototype implementation, detailing in the technical implementation of the hardware and software, and showing similarities and differences of their functional components and design decisions. Reference [24] presents a review on three functional system architectures used in self-driving platforms, comparing their construction approach and their robustness.

III. Sensing and Perception

Perception in a self-driving architecture is the stage dealing with information from sensors and turning it into meaningful data for different high-level tasks like navigation and motion planning. In this section perception will be covered in three parts: sensors, localisation and mapping, and object detection.

A. Sensors

Perception has been of interest of the field of intelligent and autonomous vehicles for more than 20 years [25], [26]. Initially, most of the developments were vision-based and applied to both infrastructure [27] and vehicles [28]. In recent years, further developments in sensors devices and processing platforms, have made possible the inclusion Lidar and Radar technologies into the suite of available sensors for self-driving applications, providing the perception stage with more data and allowing to take advantage of the strengths of each sensor technology and overcome their weaknesses. However, despite the advances in sensor technologies, there are still different challenges that need to be addressed.

As stated in the previous section, sensors in self-driving platforms are categorized into proprioceptive and exteroceptive. The former group provides the system with data regarding the position and movement of the vehicle, in an absolute reference system, like GNSS, or in a relative reference system like IMUs and encoders. When two or more of these sensors are used in conjunction, they can compensate or complement each other under adverse conditions, like losing GNSS signal, for example. Exteroceptive sensors enable the system to ”see” its surrounding environment. Cameras, lidars, radars and ultrasonic sensors generate a lot of information in form of images (2D) and point clouds (3D), from where the conditions and status of other vehicles, people and terrain can be obtained. This allows the vehicle to generate a representation of the external world, and locate itself in it in order to generate navigation plans and prevent safety-related issues. A more detailed review of sensor technology and perception in autonomous vehicles can be found in [29] and [30].

B. Localisation and Mapping

Localisation refers to the process of obtaining the relation between the vehicle state (location) and its environment, represented by a map. Two common approaches are used for doing so: 1) localisation based on proprioceptive sensors and 2) Localisation using exteroceptive sensors. In the first case, GNSS is used to provide an absolute location in a global reference system or a global map, and IMU/encoders provide relative position information, that can be used to complement or correct GNSS data when it is degraded due the vehicle been in an area where the sky is obstructed or signal is weak.

One approach used to provide additional information is to generate odometry data from range sensors like cameras and lidars, sometimes referred as visual-odometry [31]–[35] or lidar-odometry [36]–[38]. In these solutions, features and landmarks are detected and displacement is calculated from the difference between frames.

When maps are not available or provided, localisation and mapping are performed simultaneously. This technique is known as SLAM (Simultaneous Localisation And Mapping), where a map is constructed from camera or lidar data and at the same time the location of the vehicle is estimated. The main advantage of this method is that a prior map is not needed and the system could localise itself in an unknown environment. Reference [39] offers a description of localisation techniques used in autonomous platforms intended for warehouse operation and reference [40] presents a deeper review on localisation techniques for mobile robots. Irrespective of the application environment or mobile platform, those techniques and algorithms are shared with self-driving cars or larger off-road vehicles.

C. Object detection

A self-driving vehicle needs to understand its environment and be aware of all the objects and actors that could interact with it. For example, the vehicle must detect the road and the lanes within it, but it also should detect other vehicles, pedestrians, obstacles and traffic signs. Or in the case of an off-road vehicle, it also should detect types of terrain, drivable areas, livestock, trees, weather conditions, etc. All this is part of the detection stage. In addition to detection, the system should be capable of tracking the detected objects in the space domain in the time domain. The main objective is to forecast possible incidents based on predicted movements of other vehicles, people or obstacles. This generated information can also be integrated into global and local maps to improve the planning and navigation process.

Another function of detection is scene understanding which determines if an autonomous vehicle is operating within its ODD, for example in terms of the environmental conditions. And this is one of the main challenges for outdoor and off-road autonomous vehicles, to maintain their reliability across different weather conditions and handle its impact on sensor data integrity in a safe way. In [41], the authors propose a multi-feature method to perform visual localisation across different seasons throughout the year, based on scene-matching. Rain is another condition that has a big impact on cameras and lidar sensors, in particular, when droplets remain on the lenses. Reference [42] proposes a de-raining filtering method based to overcome this issue and also provides a dataset of rainy images and a method for creating synthetic water drops on top of other datasets.

IV. Data Fusion in Perception

Data fusion, also referred as multi-sensor data fusion, information fusion or sensor fusion, has received several definitions from different authors in the literature [43], [44], [45] [46], [47], but a better understanding of what it is, can be obtained by answering the following questions:

- What is involved in data fusion?
  Combine, merge or integrate homogeneous or heterogeneous data.
- What is the aim of data fusion?
  Get a better representation of a process or the environment, infer underlying information, improve the quality of the data.
- How to apply data fusion?
  Data fusion is a multi-level task, depending of the nature of the sensors, the context and the final application.

Thus, multi-sensor data fusion is a multidisciplinary field, because the information in a typical process, flows from sensors to applications, passing through stages of filtering, data enhancement and information extraction. Because of this, knowledge in a wide range of fields is required, e.g. signal processing, machine learning, probability and statistics, artificial intelligence, etc. [48].

The first step in multi-sensor data fusion is to collect and associate the data in space and time, done by performing the calibration and synchronisation of the sensors within the system. This step is extremely important as the performance of fusion stages rely on the data from different sources being consistent and referenced to a common reference frame. Some of the challenges on aggregating sensors are described by [49] where a multi-sensor fusion of camera, radar and lidar is applied to large off-road vehicles and the difficulties of time synchronisation are highlighted. Also, reference [50] describes a multi-sensing system for autonomous driving, outlining the challenges of fusing heterogeneous data from sensors and the benefits of adhering to a software architecture model based on defined interfaces and components. Further examples of recent developments in the calibration and synchronisation of sensors (cameras, lidar and/or radar) can be found in [51], [52] and [53].

As mentioned before, two types of localisation approaches can be done, based on internal sensors, or using range sensors. In the first case, the fusion of GNSS and IMU data is usually performed using techniques based on Kalman Filtering [54], [55]. This is a well-established method, and sometimes this processing is already done in the sensor, as is the case for some INSs, which embed a GNSS solution with an IMU into a single device.

In the area of visual- and lidar-odometry, there are also developments using both sensors to improve the performance compared to a unimodal solution. In [56], a method is proposed where visual-odometry is made at an initial stage, and then lidar is fused to refine the estimation, making it possible to use in indoor and outdoor conditions. Another approach is found in [57] where two robust algorithms are coupled, VISO2 for visual-odometry and LOAM for lidar- odometry. A different approach is presented in [58], where they use a multi-camera system composed of four fisheye cameras and generate virtual lidar data for fusing it into the odometry process.

A variety of SLAM techniques using different configurations of range sensors is found in the literature. There are solutions based on single cameras, multiple cameras, stereo cameras, depth cameras and 2D and 3D lidars. For example, a multi-camera SLAM system is proposed in [59], where a panoramic view is created by fusing the data from 5 cameras. The performance of the system using a single camera approach and a 3, 4 or 5 camera configuration is presented. Reference [60] presents a review on multimodal localisation approaches for mobile robots, evaluating Visual SLAM, Visual Odometry and Place Recognition in terms of the requirements to adapt to dynamic conditions and terrains, like unstructured, off-road and outdoor environments. A further review focused only on SLAM technologies is presented in [61].

All these range sensors are also integrated for the purposes of object detection and tracking. Most of the recent work in this area is based on Deep Learning and Deep networks techniques. Reference [62] proposes a lidar-camera fusion system based on Convolutional Neural Networks. They evaluate the individual performance of each sensor in addition to the fused approach. Also, in [63] is presented a Deep-Learning-based architecture for the fusion of Radar and Camera for object detection applications. Another recent work is presented in [64]: a Camera-Radar fusion for object detection using Region Proposal Networks (RPN) as a layer in a combination of networks for fusing 2D and 3D data. On the other hand, reference [65] presents an approach for fusing INS, camera and Lidar data to perform 3D object tracking based on Kalman filtering.

A further comprehensive review on multi-sensor fusion in automated driving focusing on the fusion of heterogeneous data from Camera, Lidar, Radar, Ultrasonics, GNSS, IMU and V2X communications is presented in [66].

V. Conclusions

Over the last 10 years, several advances have been made in different aspects of autonomous driving systems, from the creation of reference architectures, standards, communities, evolution of hardware and software, etc. However, in order to achieve levels 4 and 5 of autonomy, as defined by SAE standard J3016 [67], there are still different challenges that must be solved, especially in the field of perception. Developments on new lidars, radars, stereo and depth cameras, and the decreasing cost and size of these devices will allow the inclusion of several sensors per kind, opening the possibility of creating better representation of the world, but this presents challenges in terms of processing, bandwidth, synchronisation and data fusion.

In this regard, different data and information fusion techniques are being developed presently with good results, but further work must be done for taking them from isolated developments and solutions to a safety-critical self-driving platform where they should seamlessly integrate within a defined architecture or framework without affecting other subsystems. To do so, a proper methodology should be followed throughout all the product life cycle development stages: ODD definition, functional and safety requirements, architecture design, software and hardware development, and testing, verification, validation and product release.

Driving is a complex task even for humans, and we are good at dealing with unexpected situations and at making sense of multitude of multi-modal information while driving, allowing us to make good decisions and move safely through the road. However, we have to deal with stress and tiredness, and these are risk factors in non-autonomous vehicles. Through improved perception and data fusion developments, self-driving vehicles should surpass our capabilities in the pursuit of better, safer and greener transport.

ACKNOWLEDGEMENT

This work was supported, in part, by Science Foundation Ireland grant 13/RC/2094 and co-funded under the European Regional Development Fund through the Southern & Eastern Regional Operational Programme to Lero - the Science Foundation Ireland Research Centre for Software (www.lero.ie)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 754489

REFERENCES

[1] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," *IEEE Intelligent Vehicles Symposium, Proceedings*, no. Iv, pp. 163–168, 2011.

[2] S. Kato, E. Takeuchi, Y. Ishiguro, Y. Ninomiya, K. Takeda, and T. Hamada, "An Open Approach to Autonomous Vehicles," *IEEE Micro*, vol. 35, no. 6, pp. 60–68, nov 2015.

[3] P. Kruchten, "The 4+1 View Model of architecture," *IEEE Software*, vol. 12, no. 6, pp. 42–50, 1995.

[4] AUTOSAR, "Autosar standards," https://www.autosar.org/standards, Last accesed on 2020-07.

[5] M. Staron and D. Durisic, "AUTOSAR Standard," in *Automotive Software Architectures*. Cham: Springer International Publishing, 2017, pp. 81–116.

[6] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, May 2009.

[7] D. Thomas, W. Woodall, and E. Fernandez, "Next-generation ROS: Building on DDS," in *ROSCon Chicago 2014*. Mountain View, CA: Open Robotics, sep 2014.

[8] B. Gerkey, "Why ros 2?" https://design.ros2.org/articles/why_ros2.html, Last accesed on 2020-07.

[9] The Autoware Foundation, "The autoware foundation," https://www.autoware.org/, Last accesed on 2020-07.

[10] ISO, "ISO 26262 - Road vehicles – Functional safety," 2011.

[11] IEC, "IEC 61508 - Functional safety of electrical/electronic/programmable electronic safety-related systems," 2005.

[12] M. Staron and P. Johannessen, "Functional Safety of Automotive Software," in *Automotive Software Architectures*. Cham: Springer International Publishing, 2017, pp. 201–222.

[13] A. Bucaioni and P. Pelliccione, "Technical architectures for automotive systems," *Proceedings - IEEE 17th International Conference on Software Architecture, ICSA 2020*, pp. 46–57, 2020.

[14] N. Parmar, V. Ranga, and B. Simhachalam Naidu, "Syntactic Interoperability in Real-Time Systems, ROS 2, and Adaptive AUTOSAR Using Data Distribution Services: An Approach," 2020, pp. 257–274.

[15] AUTOSAR, "Autosar adaptive platform," https://www.autosar.org/standards/adaptive-platform/, Last accesed on 2020-07.

[16] S. Furst, "System/ Software Architecture for Autonomous Driving Systems," in *2019 IEEE International Conference on Software Architecture Companion (ICSA-C)*. IEEE, mar 2019, pp. 31–32.

[17] S. Kyberd, J. Attias, P. Get, P. Murcutt, C. Prahacs, M. Towlson, S. Venn, A. Vasconcelos, M. Gadd, D. De Martini, and P. Newman, "The Hulk: Design and Development of a Weather-proof Vehicle for Long-term Autonomy in Outdoor Environments," Tokyo, Japan, 2019, pp. 1–14.

[18] S. Behere and M. Törngren, "A Functional Architecture for Autonomous Driving," in *Proceedings of the First International Workshop on Automotive Software Architecture - WASA '15*. New York, New York, USA: ACM Press, 2015, pp. 3–10.

[19] ——, "A functional reference architecture for autonomous driving," *Information and Software Technology*, vol. 73, pp. 136–150, 2016.

[20] S. Akkaya, Y. Gurbuz, M. G. Zile, E. Baglayici, H. A. Seker, and A. Erdogan, "A Modular Five-Layered V-Shaped Architecture for Autonomous Vehicles," *ELECO 2019 - 11th International Conference on Electrical and Electronics Engineering*, pp. 850–854, 2019.

[21] R. Cuer, L. Piétrac, E. Niel, S. Diallo, N. Minoiu-Enache, and C. Dang-Van-Nhan, "A formal framework for the safe design of the Autonomous Driving supervision," *Reliability Engineering and System Safety*, vol. 174, no. February, pp. 29–40, 2018.

[22] M. Reke, D. Peter, J. Schulte-Tigges, S. Schiffer, A. Ferrein, T. Walter, and D. Matheis, "A self-driving car architecture in ROS2," in *2020 International SAUPEC/RobMech/PRASA Conference, SAUPEC/RobMech/PRASA 2020*. IEEE, jan 2020, pp. 1–6.

[23] C. Berger and M. Dukaczewski, "Comparison of architectural design decisions for resource-constrained self-driving cars-A multiple case-study," in *Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft fur Informatik (GI)*, vol. P-232, 2014, pp. 2157–2168.

[24] O. S. Tas, F. Kuhnt, J. M. Zollner, and C. Stiller, "Functional system architectures towards fully automated driving," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, vol. 2016-Augus, no. Iv. IEEE, jun 2016, pp. 304–309.

[25] E. Dickmanns, "The development of machine vision for road vehicles in the last decade," in *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 1. IEEE, 2002, pp. 268–281.

[26] U. Nunes, C. Laugier, and M. M. Trivedi, "Guest Editorial Introducing Perception, Planning, and Navigation for Intelligent Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 3, pp. 375–379, sep 2009.

[27] A. Broggi, K. Ikeuchi, and C. E. Thorpe, "Special issue on vision applications and technology for intelligent vehicles: part I-infrastructure," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 2, pp. 69–71, jun 2000.

[28] ——, "Special issue on vision applications and technology for intelligent vehicles: Part II - vehicles [Editorial]," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 3, pp. 133–134, sep 2000.

[29] S. Campbell, N. O'Mahony, L. Krpalcova, D. Riordan, J. Walsh, A. Murphy, and C. Ryan, "Sensor Technology in Autonomous Vehicles : A review," in *29th Irish Signals and Systems Conference, ISSC 2018*. IEEE, jun 2018, pp. 1–4.

[30] J. Zhao, B. Liang, and Q. Chen, "The key technology toward the self-driving car," *International Journal of Intelligent Unmanned Systems*, vol. 6, no. 1, pp. 2–20, 2018.

[31] B. Zhao, T. Hu, and L. Shen, "Visual odometry - A review of approaches," *2015 IEEE International Conference on Information and*

Automation, ICIA 2015 - In conjunction with 2015 IEEE International Conference on Automation and Logistics, no. August, pp. 2569–2573, 2015.

[32] Q. Lin, X. Liu, and Z. Zhang, "Mobile Robot Self-LocalizationUsing Visual Odometry Based on Ceiling Vision," *2019 IEEE Symposium Series on Computational Intelligence, SSCI 2019*, pp. 1435–1439, 2019.

[33] K. S. Krishnan and F. Sahin, "ORBDeepOdometry - A feature-based deep learning approach to monocular visual odometry," *2019 14th Annual Conference System of Systems Engineering, SoSE 2019*, pp. 296–301, 2019.

[34] M. Aladem and S. A. Rawashdeh, "A Combined Vision-Based Multiple Object Tracking and Visual Odometry System," *IEEE Sensors Journal*, vol. 19, no. 23, pp. 11 714–11 720, 2019.

[35] H. Ragab, M. Elhabiby, S. Givigi, and A. Noureldin, "The Utilization of DNN-based Semantic Segmentation for Improving Low-Cost Integrated Stereo Visual Odometry in Challenging Urban Environments," *2020 IEEE/ION Position, Location and Navigation Symposium, PLANS 2020*, pp. 960–966, 2020.

[36] B. Zhou, Z. Tang, K. Qian, F. Fang, and X. Ma, "A LiDAR Odometry for Outdoor Mobile Robots Using NDT Based Scan Matching in GPS-denied environments," *2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems, CYBER 2017*, pp. 1230–1235, 2018.

[37] I. Hamieh, R. Myers, and T. Rahman, "Construction of Autonomous Driving Maps employing LiDAR Odometry," *2019 IEEE Canadian Conference of Electrical and Computer Engineering, CCECE 2019*, pp. 15–18, 2019.

[38] L. Qingqing, F. Yuhong, J. Pena Queralta, T. N. Gia, H. Tenhunen, Z. Zou, and T. Westerlund, "Edge Computing for Mobile Robots: Multi-Robot Feature-Based Lidar Odometry with FPGAs," *2019 12th International Conference on Mobile Computing and Ubiquitous Network, ICMU 2019*, pp. 54–55, 2019.

[39] C. Cronin, A. Conway, and J. Walsh, "State-of-the-art review of autonomous intelligent vehicles (AIV) technologies for the automotive and manufacturing industry," *30th Irish Signals and Systems Conference, ISSC 2019*, pp. 1–6, 2019.

[40] S. Campbell, N. O'Mahony, A. Carvalho, L. Krpalkova, D. Riordan, and J. Walsh, "Where am I? Localization techniques for Mobile Robots A Review," *2020 6th International Conference on Mechatronics and Robotics Engineering, ICMRE 2020*, pp. 43–47, 2020.

[41] Y. Qiao, C. Cappelle, and Y. Ruichek, "Visual Localization across Seasons Using Sequence Matching Based on Multi-Feature Combination," *Sensors*, vol. 17, no. 11, p. 2442, oct 2017.

[42] H. Porav, T. Bruls, and P. Newman, "I Can See Clearly Now: Image Restoration via De-Raining," in *2019 International Conference on Robotics and Automation (ICRA)*, vol. 2019-May. IEEE, may 2019, pp. 7087–7093.

[43] F. E. White, "Data Fusion Lexicon," *The Data Fusion Subpanel of the Joint Directors of Laboratories, Technical Panel for C3*, vol. 15, no. 0704, p. 15, 1991.

[44] R. Luo, "Multisensor fusion and integration: approaches, applications, and future research directions," *IEEE Sensors Journal*, vol. 2, no. 2, pp. 107–119, apr 2002.

[45] R. C. Luo, C. C. Chang, and C. C. Lai, "Multisensor Fusion and Integration: Theories, Applications, and its Perspectives," *IEEE Sensors Journal*, vol. 11, no. 12, pp. 3122–3138, dec 2011.

[46] W. Elmenreich, "A Review on System Architectures for Sensor Fusion Applications," in *Software Technologies for Embedded and Ubiquitous Systems*, R. Obermaisser, Y. Nah, P. Puschner, and F. J. Rammig, Eds. Santorini Islands, Greece: Springer, 2007, pp. 547–559.

[47] H. Boström, S. Andler, and M. Brohede, "On the definition of information fusion as a field of research," University of Skövde, Tech. Rep., 2007.

[48] G. Velasco-Hernandez, "Multisensor Architecture for an Intersection Management System," Universidad del Valle, Tech. Rep., 2019.

[49] D. J. Yeong, J. Barry, and J. Walsh, "A Review of Multi-Sensor Fusion System for Large Heavy Vehicles Off Road in Industrial Environments," in *ISSC*, 2020.

[50] J. P. Giacalone, L. Bourgeois, and A. Ancora, "Challenges in aggregation of heterogeneous sensors for Autonomous Driving Systems," *SAS 2019 - 2019 IEEE Sensors Applications Symposium, Conference Proceedings*, pp. 3–7, 2019.

[51] H. Hu, J. Wu, and Z. Xiong, "A soft time synchronization framework for multi-sensors in autonomous localization and navigation," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, vol. 2018-July, pp. 694–699, 2018.

[52] J. Domhof, K. F. Julian, and K. M. Dariu, "An extrinsic calibration tool for radar, camera and lidar," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2019-May, pp. 8107–8113, 2019.

[53] L. Yang and R. Wang, "HydraView : A Synchronized 360 -View of Multiple Sensors for Autonomous Vehicles," pp. 53–61, 2020.

[54] S. Panzieri, F. Pascucci, and G. Ulivi, "An outdoor navigation system using GPS and inertial platform," *IEEE/ASME Transactions on Mechatronics*, vol. 7, no. 2, pp. 134–142, 2002.

[55] Wahyudi, M. S. Listiyana, Sudjadi, and Ngatelan, "Tracking Object based on GPS and IMU Sensor," in *2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*. IEEE, sep 2018, pp. 214–218.

[56] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 2174–2181, 2015.

[57] M. Yan, J. Wang, J. Li, and C. Zhang, "Loose coupling visual-lidar odometry by combining VISO2 and LOAM," *Chinese Control Conference, CCC*, pp. 6841–6846, 2017.

[58] Z. Xiang, J. Yu, J. Li, and J. Su, "ViLiVO: Virtual LiDAR-Visual Odometry for an Autonomous Vehicle with a Multi-Camera System," *IEEE International Conference on Intelligent Robots and Systems*, pp. 2486–2492, 2019.

[59] Y. Yang, D. Tang, D. Wang, W. Song, J. Wang, and M. Fu, "Multi-camera visual SLAM for off-road navigation," *Robotics and Autonomous Systems*, vol. 128, p. 103505, 2020.

[60] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. A. Velasco-Hernandez, D. Riordan, and J. Walsh, "Adaptive multimodal localisation techniques for mobile robots in unstructured environments :A review," in *IEEE 5th World Forum on Internet of Things, WF-IoT 2019 - Conference Proceedings*. IEEE, apr 2019, pp. 799–804.

[61] A. Singandhupe and H. La, "A Review of SLAM Techniques and Security in Autonomous Driving," *Proceedings - 3rd IEEE International Conference on Robotic Computing, IRC 2019*, no. 19, pp. 602–607, 2019.

[62] G. Melotti, C. Premebida, and N. Goncalves, "Multimodal deep-learning for object recognition combining camera and LIDAR data," *2020 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2020*, no. April, pp. 177–182, 2020.

[63] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A Deep Learning-based Radar and Camera Sensor Fusion Architecture for Object Detection," *2019 Symposium on Sensor Data Fusion: Trends, Solutions, Applications, SDF 2019*, 2019.

[64] Z. T. Li, M. Yan, W. Jiang, and P. Xu, "Vehicle object detection based on rgb-camera and radar sensor fusion," *Proceedings - International Joint Conference on Information, Media, and Engineering, IJCIME 2019*, pp. 164–169, 2019.

[65] A. Asvadi, P. Girão, P. Peixoto, and U. Nunes, "3D object tracking using RGB and LIDAR data," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pp. 1255–1260, 2016.

[66] Z. Wang, Y. Wu, and Q. Niu, "Multi-Sensor Fusion in Automated Driving: A Survey," *IEEE Access*, vol. 8, pp. 2847–2868, 2020.

[67] SAE, "J3016 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," 2018.