Autonomous Platform based on Small-Scale Car for Versatile Data Collection and Algorithm Verification

Jakob Valtl[1,2], Javier Mendez[1,3], Manuel Pegalajar Cuellar[3] and Vadim Issakov[1,2]

[1]Infineon Technologies AG, Germany
[2]Otto-von-Guericke Universität Magdeburg, Germany
[3]Universidad de Granada, Spain
Email: Jakob.Valtl@infineon.com

***Abstract*—In this work we present a versatile data collection and algorithm verification platform, based on an autonomous self-driving small-scale vehicle. The main advantage of the proposed demonstrator is the ability to collect data in a predefined environment without endangering traffic participants. The proposed vehicle includes multiple sensors for full 3D sensing, such as radar, LiDAR, camera and Hall-sensor. Thanks to simultaneous readout and processing, sensor data fusion and intelligent motor control is demonstrated. To store a large amount of data for real-time processing, SSD storage is connected. Finally, thanks to the optimized system power management concept, power consumption is kept low, which reduces required battery weight thus improving overall power efficiency.**

I. Introduction

Autonomous cars are an emerging research field where further research needs to be done. As a result, numerous authors are developing new AI models to improve the performance of these vehicles. However, the dataset creation for this applications is not trivial due to the difficulty to find specific scenarios that fit with the desired ones while ensuring the security and privacy of the pedestrian, vehicles, etc. in the area. During tests of sensor stability e.g. adversarial attacks, unforeseen reactions can not be excluded. Therefore, there is a need to find another approach to generate these datasets in a fast and efficient way.

At the same time, these new algorithms and models need to be tested and evaluated before a final integration in a real size vehicle. To overcome those limitations and speed up development of algorithms / models, a model car was designed based on existing open source ideas such a DonkeyCar [1]. The remote controlled (RC) car is equipped with state of the art sensors.

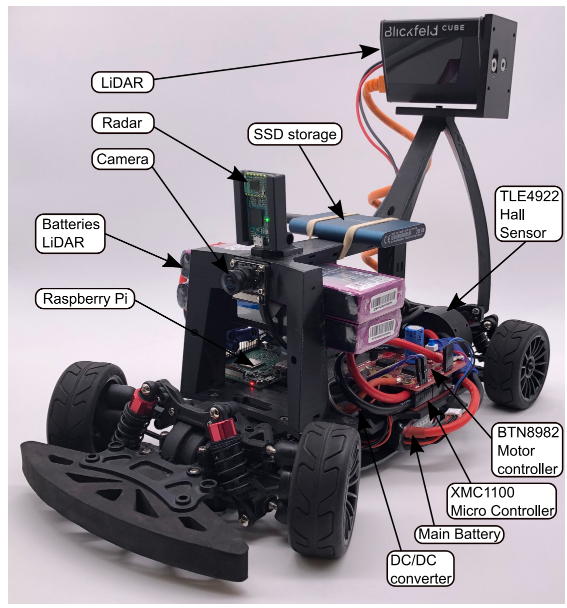


Fig. 1: Developed RC car system

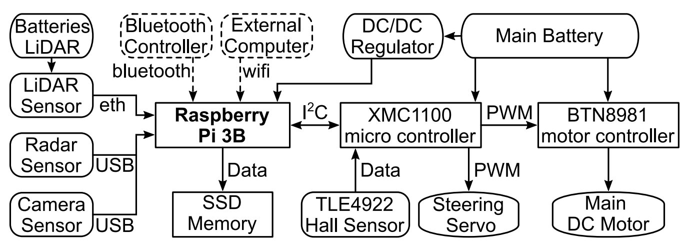


Fig. 2: System dataflow diagram

II. Prior work

The use of a reduced size vehicle in order to develop and test new algorithms has been implemented by multiple authors due to multiple reasons such as safety, security or cost efficiency. An example of this is the project of Qi Z. et al. [2] where a DonkeyCar with a camera sensor was used to successfully implement algorithms based on reinforcement learning. Similarly, T. Do et al. [3] proposed a monocular vision-based autonomous vehicle prototype using Deep Convolutional Neural Networks. This system was based on a small scale vehicle following the approach of Qi Z. et al achieving a final accuracy of 89.04% on the steering and speed decision.

However, these systems were based on a single sensor. This can lead to problems due to possible noise in the data from the camera sensor as well as vulnerabilities to adversarial attacks [4]. At the same time, camera sensor data may not be reliable when driving in dark scenarios or under bad weather conditions [5]. Therefore we propose to integrate multiple sensors leading to a sensor fusion approach where the data from different sensors will be used to complement each other.

III. System description

In the following subsections the we will explain in detail the hardware components and their interaction.

A. Mechanical Components

The base frame is a 1:10 miniature four wheel driven model car. A DC-motor as well as a servo drive, controlling the steering, are mounted on this chassis. To make room for the battery the cardan shaft was taken away. Thus only being rear wheel driven. The gearbox was modified to include a cogwheel with magnets and an Infineon TLE4922 Hall sensor to have the possibility of measuring the distance / speed of the model car. This way the motor can be PID-controlled. The structure carrying everything is 3D-printed. Infineon's micro controller XMC1100 is used to to handle all actuators. With the BTN8982 shield the DC-motor is operated.

B. Sensors

All sensors are managed with a Raspberry Pi 3B. As visual input, a 5 MP Raspberry Pi camera, with a viewing angle of 160° is used. Apart from the camera, a LiDAR and a radar sensor are included. The LiDAR sensor is the CUBE 1 from Blickfeld with a field of view of 70° x 30°, an angular resolution of up to 0.4° and a range resolution of less then 1 cm. The radar sensor is the 60 GHz BGT60TR13C from Infineon [6].

C. Data Workflow

The connection between the components is illustrated in Figure 2. There are two separate battery packs for power supply, one for all components and an extra one for the LiDAR as the LiDAR operates on a voltage level of 12 V. As the XMC1100 has its own power management system it can be operated directly with the battery while the Raspberry Pi needs an additional DCDC converter. The XMC1100 handles the Hall sensor input and communicates with the Raspberry Pi via I2C to transmit the information of the desired movement as well as to return the data from the Hall sensor. The actuators are controlled directly or indirectly through the XMC1100 by PWM signals.

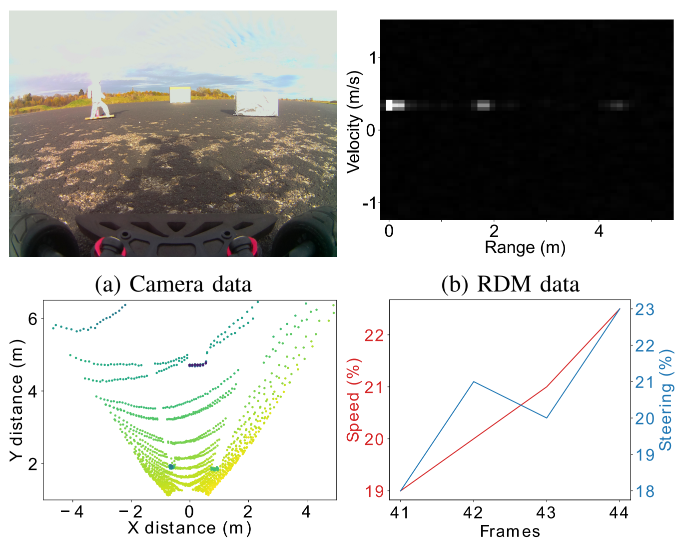
The main controller is the Raspberry Pi. It reads all sensor values, stores them to the SSD and passes orders of how to continue driving to the XMC1100. These are either generated through algorithms evaluating the sensor inputs or by a human through a bluetooth controller during the data gathering sessions. The setup and operation of the Raspberry Pi is realized by an external computer connected through WIFI.

IV. Results

Due to the optimization of the hardware communication protocols as well as the libraries to use multiple sensors, the developed system can record data of numerous scenarios using the camera, radar and LiDAR sensors simultaneously. Each of these sensors have different limitations regarding the data that can measured, i.e. LiDAR sensor has a physical constrain due to the laser reflection time. This limitation means the LiDAR sensor cannot detect / measure any target closer than 1.5 meters to the sensor. Similarly, the radar sensor, due to the specifications of the sensor, cannot detect targets further than 15 meters while static targets may be lost in a shorter range.

Multiple scenarios where tested with this system in order to ensure its proper function under different circumstances. The tested scenarios are: road intersection, roundabout, tight streets and wide streets. At the same time, different targets where used during the data recording sessions to determine the minimum size of the targets, leading to a final decision of using targets of, at least, 10x10 centimeters.

As a result of this, different datasets were created following a structure where the information of the model car, the radar data, the LiDAR data and the camera data is saved in a synchronous approach. The frame rates of the sensors have been configured based on the slowest sensor leading to a frame rate up to 20 frames (fps). The data from the internal state of the car (speed, steering, timestamp) is stored as a JSON file, the camera sensor data is saved as JPG images, the LiDAR data is stored as 3D points and finally the radar data is stored as raw data that can be later used to generate Range-Doppler-Maps (RDM) as shown in Figure 3.



(a) Camera data (b) RDM data

(c) LiDAR data (d) Speed/Angle data

Fig. 3: Frame sample of camera, radar, LiDAR and Speed/Steering data.

V. Conclusion

A fully working vehicle has been designed. It is capable of gathering data from different sensors with a frame rate up to 20 fps. Due to the reduced dimensions, in comparison to a real car, an adaptation of the environment to specific applications can easily be achieved.

At the same time, this system can be used to develop and test algorithms for autonomous vehicles in an environment where no test drivers, pedestrians or other traffic participants are put at risk, while having the benefits of working on real world instead of simulated data.

Acknowledgment

This work is a result of a collaboration between the projects ”KI-Flex” (project number 16ES1027), funded by the German Federal Ministry of Education and Research (BMBF) within the founding program Microelectronic from Germany innovation driver, and the project ”TEACHING” (project number 871385) founded by the Horizon 2020 program.

References

[1] https://www.donkeycar.com/. (Accessed: 20.10.2020).

[2] Zhang Qi and Du Tao. Self-driving scale car trained by deep reinforcement learning. *ArXiv*, abs/1909.03467, 2019.

[3] Q. Dang T. Do, M. Duong and M. Le. Real-time self-driving car navigation using deep neural network. *2018 4th International Conference on Green Technology and Sustainable Development (GTSD), Ho Chi Minh City, 2018*, abs/1909.03467:7–12, 2018.

[4] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust Physical-World Attacks on Deep Learning Models. *arXiv:1707.08945 [cs]*, April 2018. arXiv: 1707.08945.

[5] 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 Sensor Data Fusion: Trends, Solutions, Applications (SDF), Bonn, Germany, 2019*, pages 1–7, 2019.

[6] BGT60TR13C evaluation board. https://www.infineon.com/cms/de/applications/solutions/sensor-solutions/presence-detection/#!boards. (Accessed: 20.10.2020).