Sensor-based learning algorithms pave the way towards autonomous driving

Dr. Nicole Beringer, Elektrobit Automotive GmbH, Am Wolfsmantel 46, 91058 Erlangen, Germany, mail-to:nicole.beringer@elektrobit.com

Abstract

The automotive world is striving to reduce traffic fatalities to a minimum. As human errors are the main cause of accidents, the industry is pushing advanced driver assistance features. On the route towards autonomous driving, boring routine tasks for the driver will gradually become obsolete. As driver assistance features and navigation systems rely on maps, autonomous driving will need the most recent, most up-to-date maps possible. This becomes clear when investigating the limitations of the range of ego sensors or recognition algorithms as well as information, e.g. legal traffic regulations per country, which cannot be derived from sensor observations.

1 Introduction

Looking at the range of ego sensors (e.g. lidar sensors about 40m ahead) a driver may want to have the speed reduced in advance before a speed sign is reached or be warned in time to take over the control if, e.g. the autonomous driving road ends.

Similar things apply for recognition algorithms. What happens if your traffic sign is full of snow? Or if a truck hides it? Do you trust the accuracy of a recognized sign or would you feel better if the single recognition value would be improved by multiple observations and evaluated with extra-vehicle information, e.g. the set signal of the variable traffic sign?

How do you expect a vehicle sensor to find out the changing point from left hand driving to right hand driving? Or would you better trust in the system when this information comes from a source outside the vehicle well in advance?

Therefore, with road situations changing daily, in-vehicle data processing is not sufficient and needs to be enhanced with big-data learning via the cloud.

In literature you can find several publications, e.g. [4], [10], [11] as well as webinars (e.g. [12], [13], [14]) dealing with answers.

2 Quality of map data needed for ADAS applications

Before discussing the quality of map data relevant for ADAS applications, the question why ADAS need map data needs to be answered first.

In order to predict the behaviour of ADAS systems their sight horizon should have a sufficient range. Thinking of the examples for limitations of the sensor data range listed above a possible solution could be the enhancing the range by using the map data information i.e. the electronic horizon. For further details on the electronic horizon, please refer to one of the following references [1], [2], [3].

In the above example of reducing the speed the expectation of the passenger in the car usually is that the speed is reduced smoothly instead of a hard brake. If the ADAS system has to rely only on the camera, the speed limit is recognized between 50 to 100 meters in front of the car and the car can only perform a hard brake to keep the speed under the speed limit. With the electronic horizon the car gets additional information about an upcoming speed limit much earlier and can therefore perform a smooth speed reduction.

Another reason might be faulty or misinterpreted sensor data due to external factors like (partly) hidden traffic information due to snow or a truck hiding it. Another example of misinterpretations of speed limits could be with the traffic signs painted on a truck or “false friends” like 30km/h vs. 80km/h. The quality of these sensor information can be improved by using the electronic horizon.

Last but not least is to mention the example of country specific information. This information is not part of the traffic sign itself. That the car needs to change from left hand to right hand driving should be recognized earlier and smoother than by the crash detection system in order to know that the vehicle is on the wrong side of the road. Country specific information is also part of the map data.

Figure 1 shows the timeline of ADAS applications using map data.

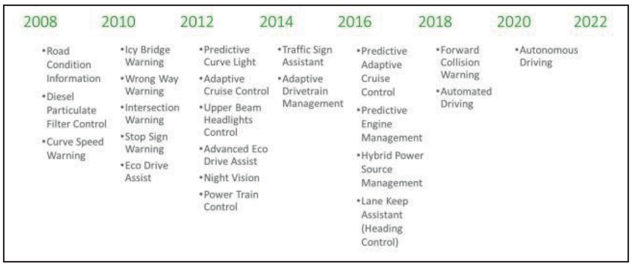

Figure 1 Timeline of ADAS applications using map data

ISBN 978-3-8007-4369-8

It is quite visible that the number of functions using map data is growing and that these functions get more and more complex. The more important it is that the data is

- Up-to-date
- Reliable and
- Precise.

To obtain and to grant these three qualities to the map data is quite challenging. The following sections show our approach to solve this challenge.

3 Architecture

In order to understand where the data is extracted from a software-architectural perspective this section will give a short introduction into Elektrobit's (further referred to as EB) architecture for autonomous driving, EB robinos [5]. This architecture is a hardware-agnostic software solution for highly automated driving. With its open interfaces, the components are interchangeable which makes it possible e.g. to switch between suppliers or to use different development modules.

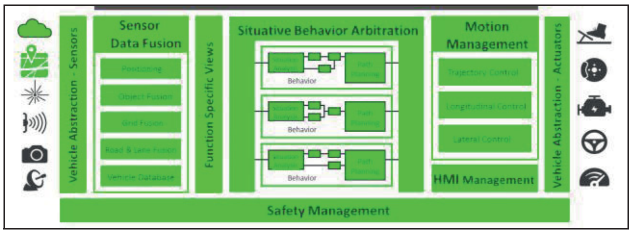

Figure 2 EB robinos high level architecture

Figure 2 shows the high level architecture of EB robinos. On the left hand side the sensors are shown. These are the input. EB robinos offers interfaces for interoceptive sensors like wheel ticks, steering angle, accelerometers or gyros, "smart" environment sensors like point clouds or object lists as well as ADASIS v2/3 [8] for map or SENSORIS [9] for cloud. A potentially usable sensor is the map data which can be combined with other sensor data, e.g. from cameras, radars, lidars etc. to obtain a more reliable interpretation of the vehicle environment. The sensor data is transferred into a sensor fusion layer to obtain an environment model of the car's surrounding, containing both static objects like lanes or traffic signs as well as other moving objects like other vehicles. In the Situative Behaviour Arbitration module the received data serves for decision taking of the vehicle actions. The result is input to the motion management layer to control the actuators. Interfaces are available for kinematic vehicle components, instrument cluster and infotainment display. A safety management module spans the whole system and integrates system health monitoring and diagnostics, safe-state triggering as well as options for redundant environment models and functions.

In terms of EB's learning approach, EB robinos offers the choice to use raw sensor data information directly or the already preprocessed information from the sensor fusion layer. This allows adapting EB robinos to the specific user's needs or demands.

With EB robinos Predictor, one of the EB robinos based products, highly accurate and up-to-date information about the road ahead for predictive driver assistance functions can be provided [6], [7] (see **Figure 3**).

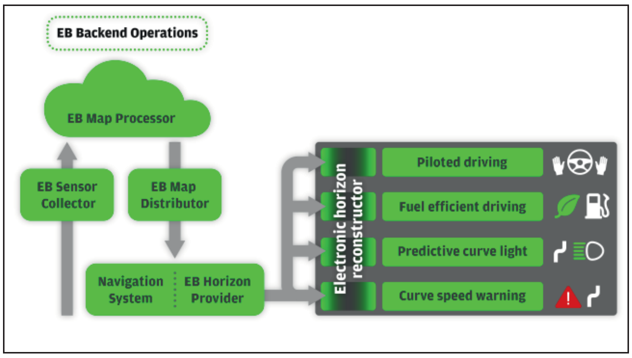

Figure 3 EB robinos Predictor using EB robinos Framework

The preparation of the data is done on a central ECU with access to an usually big data base. The data is transmitted to several ECUs via the bus network of the vehicle. The information provided is widely known as "electronic horizon".

4 Sensor-based learning approach

Based on EB's experience in driver assistance components, navigation systems, cloud infrastructure, remote and real time analytics, EB is able to enrich maps with optimizations based on its experience with sensor information.

We can use the driving vehicle itself as a sensor to continuously explore and improve the map data.

Sensor information is not primarily used to feed ADAS but is used to collect and extract features of the vehicle environment.

Sensor information of a single vehicle are limited. But integrating this information into a continuous flow of data from a fleet of cars allows to improve the precision of ADAS maps. This can be obtained due to the frequency of updates or validating and correcting the data.

With EB's sensor-based learning approach that runs in a combined way in-vehicle and in the cloud, both ego-sensor information from each contributing vehicle is covered. This approach allows to maintain the map according to the required quality features discussed in section 2.

In **Figure 4** you can see two areas. The in-car processing is displayed at the bottom of figure 4. Sensor data is collected and then sent via secure connection to the cloud.

ISBN 978-3-8007-4369-8

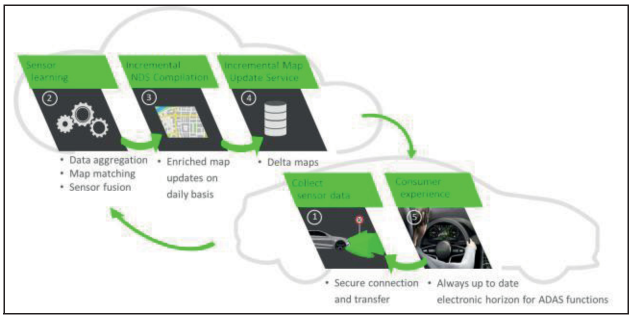

Figure 4 The learning loop step-by-step

In the cloud, each information is then aggregated, enriched and optimized with vehicle-external information from other sources like maps, legal regulations, traffic cameras, traffic lights, digital traffic sign information or similar sources.
The data is compared with a previous map version to find differences or missing attributes before an updated map is provided by the cloud. When downloaded by the vehicle this updated map feeds ADAS via the electronic horizon.

How does this work in more detail?

4.1 Sensor Data Collection

The vehicle already uses several sensors which are useful to understand the environment of the vehicle.
The cameras of the vehicles for instance are able to give insight about several kinds of "objects", e.g. traffic signs or lane markings. This information can be directly sent to the cloud and processed as it is. Please bear in mind that with lane markings the noisiness of recognized lane information far ahead of the vehicle is quite high and therefore not relevant. For the collection the information next to the car and behind the car is important.
With two-and-a-half-dimensional maps from newer cameras the top view of the road is fed with additional information about the height of each point. Both views serve as input for the in-vehicle image processing and machine learning techniques in order to extract lane markings, boundaries, arrows painted on the road, pedestrian crossings, etc.
Other sensors like gyroscopes or accelerometers allow the calculation of the curvature performed by the car, i.e. the road curvature or the movement of overtaking another car. To classify which is which, the cloud-based validation step described in the next section is relevant.
The road friction coefficients can be estimated by using ESP or ABS information. For slopes, accelerometers and other proprioceptive sensors are essential.

4.2 Cloud-based learning of Sensor Data

All relevant in-vehicle data can now be sent to the cloud for further processing. For the collection, no real-time transmission is needed, even though some limits to the delay are considered in terms of desired map update frequency, sensor quality and the kind of data to be collected. All cloud services are dynamic. The cloud architecture is scaled automatically according to the actual load [11].
The higher the requests, the more server instances will be in use and vice versa.

4.2.1 Bootstrapping base maps

The approach bootstraps HD map information with base maps of different focus (commercial maps, community maps, sensor maps, etc.) in order to capture and replay them with the most recent learned sensor attributes.
The base map is essential since our solution is not creating a map from scratch, as this would mean that ADAS could only use the meaningful part of the map after a long initialisation phase to add information to a blank map. The sensor attributes are first matched on a recent map, assigned to edges and nodes of the road network and then pre-processed ego-sensor information from single vehicles which are sent in a secure manner from EB's in-vehicle client for secure cloud connection to EB's cloud backend. The local pre-processing of the ego-sensor data cares on validating whether an information is new compared to the information it already has out of the electronic horizon. In-vehicle validation is also part of this pre-processing in order to detect errors of single sensors.

4.2.2 Anonymization of data in the cloud

The data is sent to the cloud via a communication provider that is responsible for setting up the communication channel and performing authentication. The pre-processed ego-sensor information has now to be anonymized in the cloud, i.e. all personal information needs to be separated from the sensor data. This step is necessary for data protection and privacy of the persons behind the vehicle (driver, car owner, etc.). EB takes data protection, security and privacy already into account when designing the system architecture. As a result, every learning, validation or enhancement for providing high-resolution maps is done on anonymized data.

4.2.3 Patterns for validating sensor data from vehicles

Having the data anonymized in the cloud the validation of the sensor data has to be done in order to filter out error identifiers from the data.
Before the real validation starts, the data first goes through a filtering stage in order to check the validity of the timestamp and the location. The data quality in terms of GPS uncertainty or the movement profile of the data corresponds to the vehicle sending the information (e.g. car is on a ferry). Data which does not pass the first validation is discarded. The validation is based on statistical plausibility checks and/or comparison with road classification as well as runtime information of the vehicle at the time a sensor information comes up. It gets clear when taking speed limits as an example:

- Statistical plausibility: How often was a specific sign reported at this position?
- Runtime information: what speed did this vehicle have at the time of the detection of the specific sign?
- Combined runtime with statistical plausibility: which average speed do vehicles have when passing this specific sign?
- Road classification: Is this speed limit plausible on the road it is on? E.g. the recognized value from the car was 100 km/h. The street was inside a town.

Having done these steps in the learning phase, the next step is to enrich and to update already existing map information, which shall be integrated into the map update.

4.2.4 Distribute map updates from cloud to vehicle

Once the cloud has the updated map information ready, an immediate use within the vehicle would be nice. Therefore, EB robinos Predictor provides NDS snippets of the updated regions relevant for the vehicle.

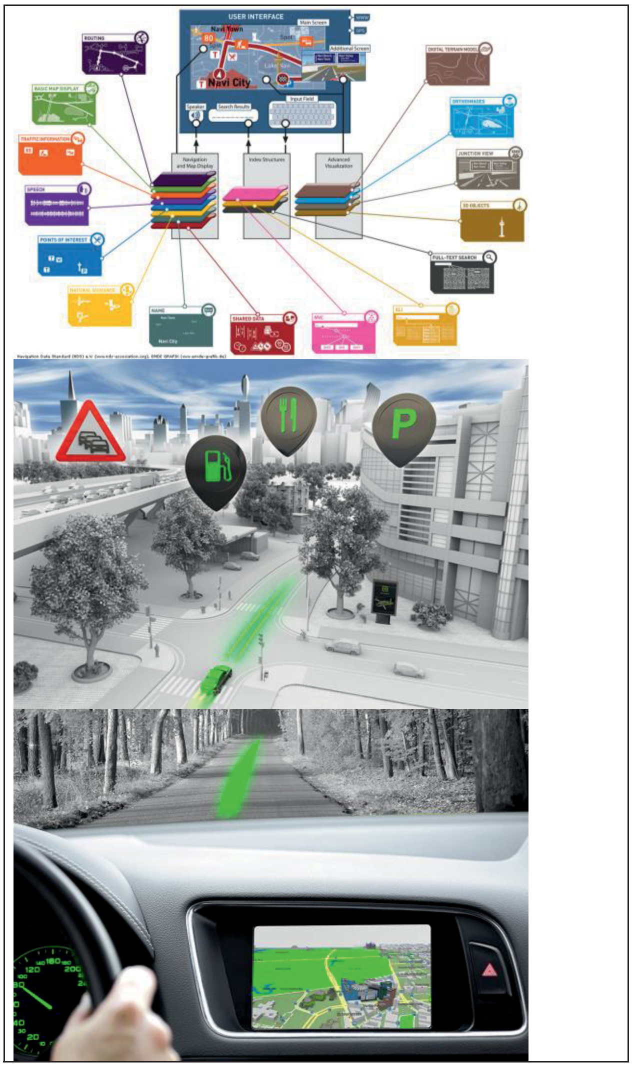

Figure 5 map distribution

This can be full updates or incremental updates. The latter updates only the actually changed tiles of the maps, the former replaces the whole map database.
Currently, the map updates are available on a daily basis.
The update mechanism downloads the updates as packages via an authenticated channel (**Figure 5**). In the vehicle, the maps are checked for transmission errors and in the positive case installed. With incremental updates, only the new parts of the database are changed.
Of course, similar processing needs to be done when talking about real-time information like hazards, accidents, weather influences on the road or other time-critical events. The difference is that this real-time information has to be available promptly. This is done using some already foreseen layers in the NDS (Navigation Data Standard [15]) data format to support driver assistance systems ("ADAS layer") and automated driving ("Auto drive layer"). The main advantage of NDS here is the provision of incremental update technologies so that the map contents with little data volume can be transferred to the vehicle. In this cycle of capturing data in the car, sending it into the cloud, map processing and enriching HD map data and sending this back to the car not only NDS as an established standard is used but also other standards for interfaces and formats in the industry: ADASIS and the future standard SENSORIS.
With EB's processing loop (**Figure 6**) which takes into account not only ego-sensor information or only extra-vehicle information the limitations listed above can be overcome.

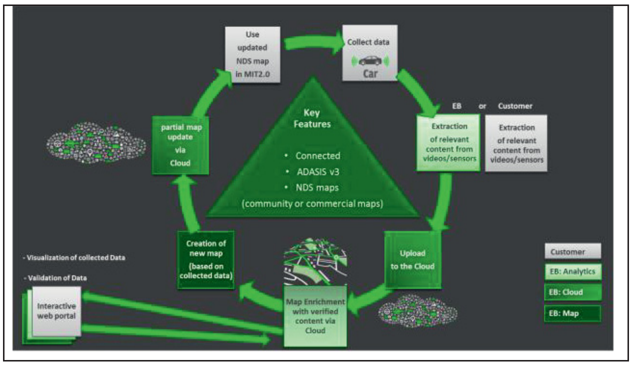

Figure 6 EB's sensor-based learning loop

The result is to have technologies to contribute to highly accurate and up-to-date information by intelligently processing of the different sensor data and to compensate errors of single ego-data by heterogeneous sources – not all vehicles will then make the same errors.

5 Examples

This section describes the different processing steps based on three examples:

- Speed limits
- Road curvature

- Lane markings and lane boundaries.

5.1 Speed limits

Figure 7 shows the processing in the car. On the left you can see where the car is driving based on the recording of the front camera. The right window will show the result of the speed sign recognizer shortly before it passes the crossing. As the recognizer correctly recognized a 70km/h sign this information is sent to the cloud including the geo-localization.

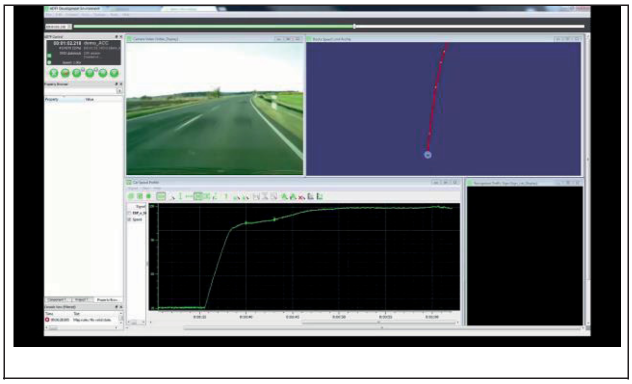

Figure 7 Speed limit recognition in the car

Figure 8 shows the processing in the cloud. The clustering that is needed to provide the exact position of the speed sign is visualized by the blue spots.
However, this is only a part of the processing. ADAS usually is not just interested in the location of the speed sign but in the content, i.e. the maximum allowed speed limit on the road. The algorithms in the cloud therefore calculate the maximum allowed velocity with the location of the traffic signs and other sources.

Figure 8 Speed limit recognition in the cloud

Comparing the output of the algorithms with the data in the base maps are shown with different line styles. This information will enter the map update with the next iteration. Finally, the age of the information is determined using another color code: the greener the sign the newer it is.

5.2 Road curvature

Figues 9 and **10** show the processing of the road curvature in the vehicle and in the cloud.
In-vehicle processing allows the estimation of the curvature of the vehicle's trajectory by means of the gyroscope, the accelerometer, the steering wheel angle and individual speed wheels. The plot shows the curvature values as estimated from each sensor.
The different sources give slightly different values which need to be aggregated in the cloud processing.

Figure 9 Road curvature estimation in the car

In contrast to the classical driver assistance functions, the slight differences in the in-vehicle values are not an issue, because the cloud-processing easily identifies such cases as outliers.
Figure 10 shows many tracks collected on a road section. The raw measurement values obtained from single vehicles are interpolated to find out the real road curvature. Apart from this these events are statistically irrelevant.

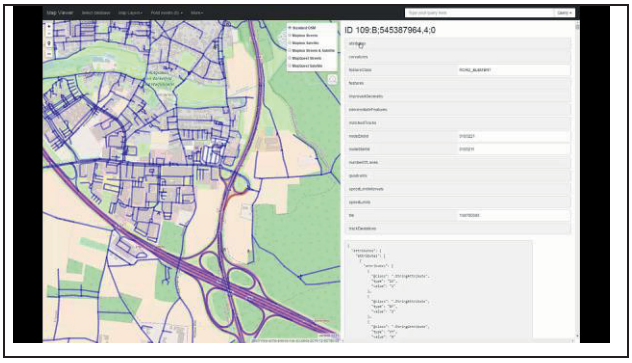

Figure 10 Road curvature in the cloud

5.3 Lane markings and lane boundaries

The last example shown in **Figure 11** shows the lane marking pre-processing. Starting from the 2.5d maps, lane markings are extracted from the road to identify the lane boundaries.
The comparison with geo-localized satellite pictures shows the accurateness of this method. The same kind of

ISBN 978-3-8007-4369-8

processing can be done also without a 2.5d map, but using the data from a typical series camera.

Figure 11 Lane markings and lane boundaries

6 Conclusion and Outlook

The technology presented in this paper can be seen as the basics to provide highly accurate, up to date maps for the development of self-driving cars. It allows to meet the quality criteria defined in section 2:

- Up-to-date-ness: data is continuously transferred to the cloud. Any changes to the road for instance are detected as soon as the vehicle passes by. The sensor quality may need to be confirmed by other vehicles.
- Reliability: deviations of map contents used in the vehicle are detected by the vehicle sensors and directly submitted to the cloud. The data is always up-to-date and with this the probability of unconscious changes is drastically reduced as other vehicles always crosscheck the map content as well.
- Precision: The map becomes more precise with more collected data.

It has been shown that with our machine learning approach for predictive driving, the named limitations of ego-sensor ranges and recognition algorithms can be overcome to pave the way towards autonomous driving.

7 Literature

[1] Ludwig, Jürgen: Elektronischer Horizont: Vorausschauende Sicherheitssysteme und deren Anbindung an Navigationssysteme, ATZ Elektronik 2012

[2] Ludwig, Jürgen: Electronic horizon – efficiency, comfort and safety with map data, ATZ Elektronik 2014

[3] Ludwig, Jürgen: Electronic horizon – flexible implementation of predictive driver assistance features, Tech Paper 2015

[4] Jesorsky, Oliver: Kartendaten – Mehrwert für Assistenzsysteme, Hanser Automotive 2011

[5] www.open-robinos.com

[6] https://www.elektrobit.com/products/eb-robinos/predictor/

[7] Barthel, Thomas: Map data as co-pilot: extended scope for autonomous driving, Embedded Computing Design, 2017

[8] www.ertico.com/adasisforum

[9] http://erticonetwork.com/ertico-coordinate-standard-development-vehicle-cloud-data/

[10] Holve, Rainer; Dormann, Holger: Big Data Analytics in der Automobilindustrie, ATZ Elektronik 2016

[11] Schneider, Tobias: Achieving cloud scalability with microservices and DevOps for connected cars, eetimes, 2016

[12] Dormann, Holger: Map updates for autonomous driving – the sensor cloud approach, Automotive World Webinar, 2015

[13] Labella, Thomas; Beringer, Nicole: Sensor based learning - one step closer to automated driving, automotive megatrends.com, 2016

[14] De Rooij, Frans, Tremeer, Barry: Enabling cars to see beyond their sensors, Automotive World Webinar, 2015

[15] www.nds-association.org/