Providentia - A Large Scale Sensing System for the Assistance of Autonomous Vehicles

Annkathrin Krämmer[*12], Christoph Schöller[*12], Dhiraj Gulati[12] and Alois Knoll[2]

[1]fortiss GmbH, Munich, Germany
[2]Technical University of Munich, Munich, Germany
[*]These authors contributed equally to this work

Abstract—The environmental perception of autonomous vehicles is not only limited by physical sensor ranges and algorithmic performance, but also occlusions degrade their understanding of the current traffic situation. This poses a great threat for safety, limits their driving speed and can lead to inconvenient maneuvers that decrease their acceptance. Intelligent Transportation Systems can help to alleviate these problems. By providing autonomous vehicles with additional detailed information about the current traffic in form of a digital model of their world, i.e. a digital twin, an Intelligent Transportation System can fill in the gaps in the vehicle's perception and enhance its field of view. However, detailed descriptions of implementations of such a system and working prototypes demonstrating its feasibility are scarce. In this work, we propose a hardware and software architecture to build such a reliable Intelligent Transportation System. We have implemented this system in the real world and show that it is able to create an accurate digital twin of an extended highway stretch. Furthermore, we provide this digital twin to an autonomous vehicle and demonstrate how it extends the vehicle's perception beyond the limits of its on-board sensors.

I. INTRODUCTION

The environmental perception and resulting scene and situation understanding of autonomous vehicles is severely restricted by limited sensor ranges and object detection performance. Even in the vicinity of a vehicle, occlusions lead to incomplete information about its environment. The resulting uncertainties pose a safety threat for itself and other traffic participants. To operate safely, driving speed must be reduced, which slows down traffic. Furthermore, the convenience of using autonomous vehicles can be reduced, as the vehicle must spontaneously react to unforeseen scenarios. This can result in abrupt breaking and adjustment maneuvers.

Intelligent Transportation Systems (ITS) can alleviate these problems by providing autonomous vehicles – as well as legacy vehicles or drivers – with additional information about every traffic participant and the overall traffic situation [1, 2]. In particular, an ITS can sense the traffic from superior perspectives and with an extended coverage compared to an individual vehicle. Providing a vehicle with this additional information leads to a better understanding of its surrounding scene and enables it to plan its maneuvers more safely and conveniently. Furthermore, an ITS with described capabilities allows to implement a multitude of additives services to further support decision making.

However, building such a system is a challenging task that ranges from the choice of the right hardware and sensors, to their optimal deployment and utilization in a complex software stack. The perception of an ITS must be reliable and robust with respect to different weather, light and traffic density conditions. Such reliability can only be guaranteed with a combination of sensors of different modalities, redundant road coverage with overlapping field of views (FoV), accurate calibration [3] and robust detection and data fusion algorithms.

While we outlined ideas of how such a system could be designed in prior work [4], in this work we propose a concrete, scalable architecture. This architecture is the result of our real world build-up experience of the ITS Providentia. It includes the system's hardware, as well as the software to operate it. In particular, for hardware we discuss the choice of sensors, the deployment of edge computing for the fast and distributed processing of heavy sensor loads and our network architecture. We outline our software stack and the detection and fusion algorithms used to generate an accurate and consistent model of the world, which we call the digital twin. The digital twin includes information such as the position, velocity, type and a unique identifier for every observed vehicle. By providing this digital twin to an autonomous vehicle, we demonstrate that it can be used to extend the perception of the vehicle beyond the limits of its on-board sensors.

II. RELATED WORK

With the emergence of autonomous driving, the need for ITS to support autonomous vehicles is continuously increasing. Hence, many projects with the goal to develop prototypical ITS have been initiated. However, their goals differ strongly.

One aspect is vehicle-to-everything (V2X) communication, a necessary component to transmit information from an ITS to the vehicles. The research project DIGINETPS [5] sets a strong focus on communication topics, but also detects pedestrians and cyclists for intersection management and communicates traffic signals to vehicles. Similarly, the project Veronika [6] focuses on communication between vehicles and traffic signals to reduce emissions and energy consumption. On the other hand, the Testfeld Autonomes Fahren Baden-Württemberg [7] aims to develop a system that focuses on providing information for testing and evaluation of autonomous driving functions by capturing the traffic with multiple sensors.

Despite the number of initiated projects, the literature about the design and replication of such systems in practice, is sparse. Most contributions focus either on communication aspects of ITS – often from a conceptual point of view [8, 9] – or on the development of methods that make use of such a system. Examples of such methods are traffic density prediction[10, 11], danger recognition[12], vehicle motion prediction [13], and vehicle re-identification [14, 15].

Contrary to the described systems and literature, our work focuses on the system architecture and implementation of an ITS as a whole. The system we describe has the primary purpose of extending the vehicles' perception with a far-reaching view to improve their scene and situation understanding.

III. THE PROVIDENTIA SYSTEM ARCHITECTURE

Providentia is a distributed sensor system, consisting of multiple edge computing nodes, a complex software architecture and a broad range of state-of-the-art algorithms. We have built it as a prototype on the German highway A9 near Munich to provide a digital twin of the current traffic during any time and day of the year. In this section we describe the design of our system. We begin with the hardware and software setup and then explain the detection and fusion algorithms used.

A. Hardware and Software Setup

To build our system we have equipped two gantry bridges with a distance of approximately $440\,\text{m}$ with sensors and computing hardware. Each gantry bridge represents one measurement point and is depicted in Fig. 1. To achieve a high perception robustness, we use sensors of different measurement modalities and cover the whole stretch between our measurement points redundantly. The overall setting of our system with the redundant coverage of the highway is illustrated in Fig. 2.

Each measurement point comprises eight sensors with two cameras and two radars per viewing direction. In particular, in one direction one radar covers the right side and the other one the left side of the highway. The cameras have different focal lengths in order to capture the far and near range. The combination of sensors with different modality ensures good detection results in all weather, light and traffic conditions. Besides the redundant coverage with the sensors on one measurement point, we selected the positions of the two measurement points in such a way that their overall FoVs overlap as well. This further increases redundancy and thus robustness. The coverage of the highway stretch from different viewing directions helps to resolve sensor failure and occlusions, and it allows smooth transitions while tracking vehicles through all sensor FoVs.

The radars we use are specialized traffic monitoring radars, manufactured by SmartMicro, and cameras of type Basler acA1920-50gc. All sensors of one measurement point are connected to a Data Fusion Unit (DFU), which serves as a local edge computing unit and runs with Ubuntu 16.04 Server. It is equipped with two INTEL Xeon E5-2630v4 $2.2\,\text{GHz}$ CPUs with $64\,\text{GB}$ RAM and two NVIDIA Tesla V100 SXM2 GPUs. All sensor measurements from the cameras and radars

Fig. 1: One of our measurement points on the highway A9. All sensor information is processed by the local Data Fusion Unit. The two radars directed towards north are installed on the other side of the gantry bridge and hence not visible from this perspective.
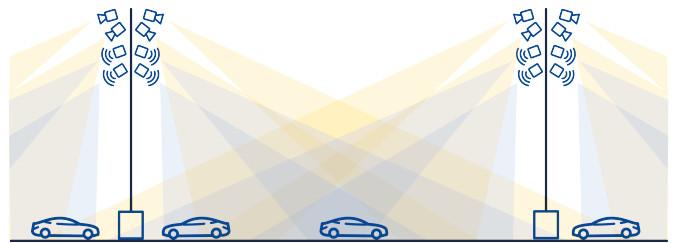

Fig. 2: Schematic illustration of the Providentia sensor setup with overlapping FoVs for redundancy.

are fed into our detection and data fusion toolchain running on this edge computing unit. This results in object lists, containing all tracked traffic participants in the FoV of that measurement point. Each DFU sends this object list to a backend machine, where they are finally fused into the digital twin that covers the whole observed highway stretch.

Our full architecture is depicted in Fig. 3. For seamless connectivity we use ROS on all the nodes. The final digital twin is either communicated to autonomous vehicles, or to a frontend where it can be visualized appropriately for drivers or an operator.

B. Object Detection and Data Fusion

The first step to create a digital twin of the highway is to detect all vehicles in the sensors' measurements. While we use pre-installed firmware for object position and velocity detection with our radars, the vehicles in the camera images
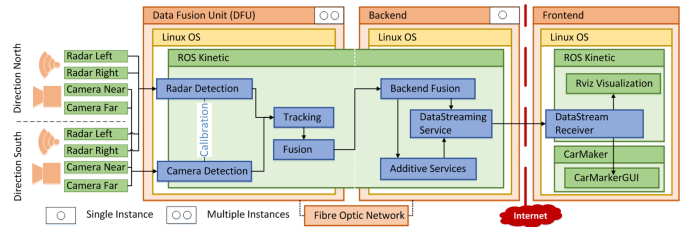

Fig. 3: Platform architecture of the Providentia system.

are detected on our DFUs. For this purpose, we leverage the YOLOv3 [16] object detector. In addition to regressing bounding boxes with a confidence score, this detector classifies detected vehicles in types like car and truck. To compute the 3D positions of the vehicles from the detected bounding boxes, we shoot a ray through the bounding box and intersect it with the street-level ground plane.

Then we transform the resulting vehicle detections, along with the detections of the radars, into a common coordinate system before we fuse them into a consistent world model. To make all this possible, a precise calibration of all sensors and the measurement points is necessary. While we intrinsically calibrated the cameras individually with the common checkerboard method before installation, the overall extrinsic calibration of the system is non-trivial. Our system has not only a high number of sensors and degrees of freedom, but also makes use of sensors with heterogeneous measurement principles. We address these challenges by using our radars' in-built calibration algorithms, vanishing point methods [17] and manual fine-tuning, i.e. manually minimizing re-projection errors. To calibrate our whole system with respect to the world (GPS coordinates), we use a GPS device and information from a high definition map.

Concerning the sensor data fusion, a large-scale system like ours poses many challenges. On the highway we can observe a very high number of vehicles that need to be tracked in real time. Therefore, the data fusion system should scale for over one thousand vehicles. Besides that, the number of targets is unknown and our fusion must be robust with respect to clutter and detection failures. Conventional filtering methods handling every observed vehicle separately, e.g. multiple Kalman filters or Multiple Hypotheses Tracking [18], require to explicitly solve a complex association problem between the system's sensor detections and tracked vehicles. This severely limits scalability. Therefore, we make use of the Random Finite Set (RFS) framework [19, 20], more precisely the Gaussian Mixture Probability Hypothesis Density (GM-PHD) filter [21]. This filter avoids the explicit data association step and has shown to balance our runtime and scalability constraints. Additionally, it handles time-varying target numbers, clutter and detection uncertainty within the filtering recursion.

We add tracking capabilities to our GM-PHD filter by extending it with ideas from Panta et al. [22]. In particular, we make use of the tree structure that naturally arises in the GM-PHD filter recursion and appropriate track management methods. For motion and sensor models, we use a standard constant velocity kinematic model and a zero-mean Gaussian white noise observation model, respectively. We empirically tuned all parameters for our sensor and scenario specifications. To fuse the data from different sensors and measurement points, we adapted the method from Vasic et al. [23] that is based on Generalized Covariance Intersection [24]. In order to ensure the scalability of our system setup, we implement a hierarchical data fusion concept, where we first perform a local sensor fusion at each measurement point that leads to vehicle tracklets. A second-level fusion of all measurement

(a) Camera Image
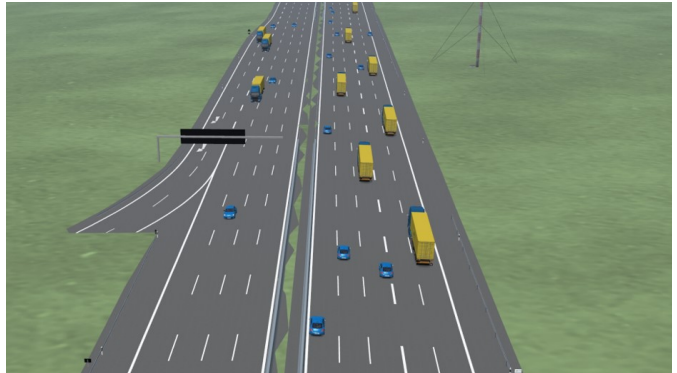
(b) Digital twin

Fig. 4: Qualitative example for how our system captures the real world (a) in a digital twin (b). We recreate the scene with generalized 3D models for different vehicle types for visualization purposes. During operation, all information is sent to the autonomous vehicle in form of an object list. Note that our system resolves the occlusion of the truck and car in the top-left by fusing multiple sensor perspectives.

point results is then performed in the backend node. This step generates a consistent model of the whole highway scene that is covered by our system.

Our fusion concept allows an easy extension of our system, because each measurement point is set up independently of the others and integrated the same way in the backend. The resulting digital twin comprises the position, velocity and type for every observed vehicle, each one having a unique tracking identifier. It can be used to implement further additive services, for example motion prediction for each vehicle, congestion recognition, lane recommendations, and collision warnings.

IV. QUALITATIVE RESULTS

In this section we analyze the object recognition performance of our system. In particular, at first we evaluate the ability of our system to capture the highway traffic, and then we demonstrate its potential to extend an autonomous vehicle's perception of the scene. We restrict ourselves to a qualitative evaluation, but plan to add quantitative results in the future. A quantitative evaluation is difficult due to a lack of sufficient ground truth. As described in Sec. III, our system
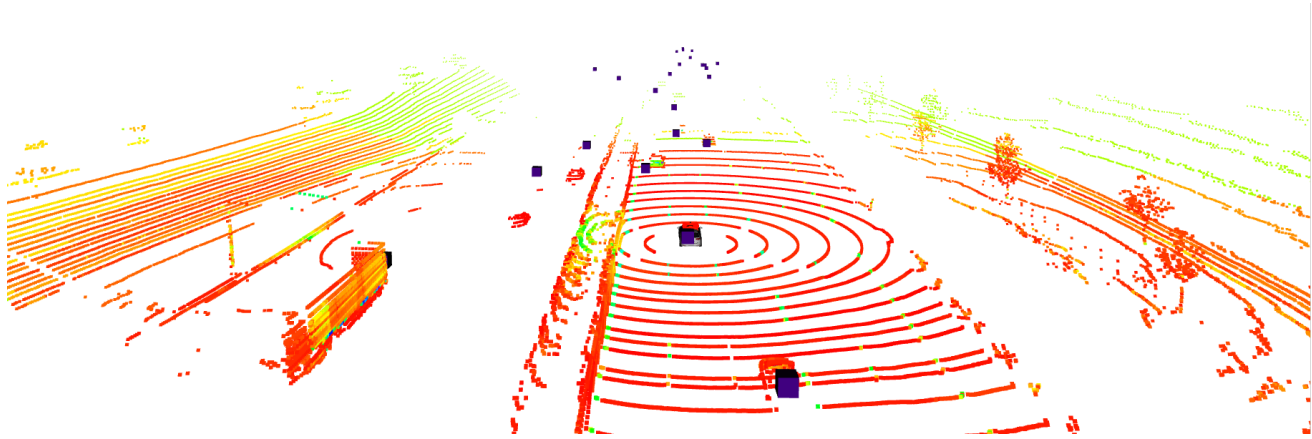
Fig. 5: An autonomous vehicle driving through our Providentia system. The dots visualize its lidar measurements and the purple cubes represent the vehicles perceived with our Providentia system. While the vehicle's own lidar range is severely limited, its perception and resulting scene understanding gets extended into the far distance with information from our system.

redundantly covers a stretch of about $440\,\mathrm{m}$ of the road, this corresponds to the distance between our measurement points. All our examples presented in this section were captured on this highway stretch under real-world conditions.

In Fig. 4 we show an example of the digital twin of the current traffic on the highway that our system computes. It is a visualization of the information that also gets sent to autonomous vehicles to extend their perception. Our system is able to reliably detect the vehicles on the road. Two occluded vehicles were successfully detected at the beginning of the highway exit. This is only possible by making use of multiple sensor perspectives and fusing them. Only a single truck was misclassified as car in the back of the scene. Note that currently the visualization of the highway stretch is slightly misaligned. In future we want improve this by making use of our high definition map. Even though our system is not optimized yet, we are able to reliably capture the observed road stretch with a frequency of $5\,\mathrm{Hz}$.

We also transmit this digital twin to an autonomous vehicle to extend its environmental perception and situation understanding. Vehicles perceive their environment with lidars whose measurement ranges are severely limited, and the point cloud density in the distance becomes increasingly sparse. Vehicular cameras can capture a more distant environment compared to lidars, but objects that are too far away appear small on the image and will not be reliably detected. Furthermore, the vehicle's low perspective is prone to occlusions. Fig. 5 shows how an autonomous vehicle driving through our system perceives its environment. While the effective range of the lidar ends at less than $60\,\mathrm{m}$, it receives the digital twin from our system. Each vehicle detected by our system is represented by a violet cube. This additional information extends its environmental perception to up to $400\,\mathrm{m}$. In principle, a system such as ours can extend the perception of the vehicle even further since we designed it with scalability in mind. The maximum distance is only limited by the number of the built-up measurement points.

We also tested our systems under harsh environmental conditions as shown in Fig. 6. Despite the heavy snow storm, our traffic radars, as well as the object detection algorithm for the cameras deliver reliable results. This is important, such that autonomous vehicles can always rely on the additional information they receive from our system.

V. CONCLUSION

To improve the safety and comfort of autonomous vehicles, one should not only rely on on-board sensors, but extend their perception and resulting scene understanding with additional information provided by modern ITS. With their superior sensor perspectives and spatial distribution, ITS can provide information far beyond the perception range of an individual vehicle. This can resolve occlusions and lead to a better long term planning of the vehicle.

While research on specific components and use-cases of ITS is popular, information on building up such a system as a whole is sparse. In this work we have described how a successful modern ITS can be designed. This includes our hardware and sensor setup, recognition algorithms and a data fusion concept. We have shown that our system is able to achieve reasonable results at capturing the traffic on the observed highway stretch and can generate a reliable digital
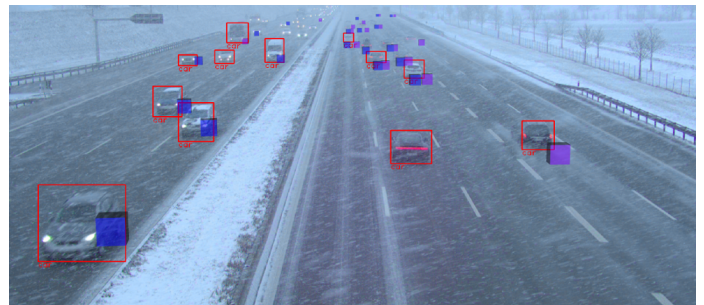

Fig. 6: Our system detects vehicles even under harsh weather conditions. The blue and violet cubes represent detections of two different traffic radars and the red bounding boxes are camera detections.

twin in near real-time. We have further demonstrated that it is possible to integrate the information captured by our system into the environmental model of an autonomous vehicle to extend its limited perception range.

In future we plan to conduct a quantitative evaluation of our system performance and we would like to enrich the information provided to vehicles with a local high definition map. Furthermore, it would be interesting to investigate further additive services benefiting autonomous vehicles that can be realized with our system.

REFERENCES

[1] Kashif N. Qureshi and Hanan Abdullah. A survey on intelligent transportation systems. *Middle-East Journal of Scientific Research (MEJSR)*, 2013.

[2] Hamid Menouar, Ismail Guvenc, Kemal Akkaya, A. Selcuk Uluagac, Abdullah Kadri, and Adem Tuncer. UAV-enabled intelligent transportation systems for the smart city: Applications and challenges. *IEEE Communications Magazine*, 2017.

[3] Christoph Schöller, Maximilian Schnettler, Annkathrin Krämmer, Gereon Hinz, Maida Bakovic, Müge Güzet, and Alois Knoll. Targetless rotational auto-calibration of radar and camera for intelligent transportation systems. *arXiv*, 2019.

[4] Gereon Hinz, Martin Büchel, Frederik Diehl, Guang Chen, Annkathrin Krämmer, Juri Kuhn, Venkatnarayanan Lakshminarasimhan, Malte Schellmann, Uwe Baumgarten, and Alois Knoll. Designing a far-reaching view for highway traffic scenarios with 5G-based intelligent infrastructure. In *8. Tagung Fahrerassistenz*, 2017.

[5] Diginetps - the digitally connected protocol track. URL http://diginet-ps.de.

[6] Veronika. URL https://veronika.uni-kassel.de.

[7] Testfeld Autonomes Fahren Baden-Württemberg. URL https://taf-bw.de.

[8] Jeffrey Miller. Vehicle-to-vehicle-to-infrastructure (V2V2I) intelligent transportation system architecture. In *Intelligent Vehicles Symposium (IV)*, 2008.

[9] Igor Kabashkin. Reliability of bidirectional V2X communications in the intelligent transport systems. In *Advances in Wireless and Optical Communications (RTUWO)*, 2015.

[10] Shanghang Zhang, Guanhang Wu, Joao P. Costeira, and Jose M. F. Moura. FCN-rLSTM: Deep spatio-temporal neural networks for vehicle counting in city cameras. In *International Conference on Computer Vision (ICCV)*, 2017.

[11] Shanghang Zhang, Guanhang Wu, Joao P. Costeira, and Jose M. F. Moura. Understanding traffic density from large-scale web camera data. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[12] Lijun Yu, Dawei Zhang, Xiangqun Chen, and Alexander Hauptmann. Traffic danger recognition with surveillance cameras without training data. In *International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2018.

[13] Frederik Diehl, Thomas Brunner, Michael Truong Le, and Alois Knoll. Graph neural networks for modelling traffic participant interaction. *arXiv*, 2019.

[14] Yantao Shen, Tong Xiao, Hongsheng Li, Shuai Yi, and Xiaogang Wang. Learning deep neural networks for vehicle re-id with visual-spatio-temporal path proposals. In *International Conference on Computer Vision (ICCV)*, 2017.

[15] Yi Zhou and Ling Shao. Viewpoint-aware attentive multi-view inference for vehicle re-identification. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[16] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[17] Neeraj K. Kanhere and Stanley T. Birchfield. A taxonomy and analysis of camera calibration methods for traffic monitoring applications. *Transactions on Intelligent Transportation Systems (T-ITS)*, 2010.

[18] Samuel S. Blackman. Multiple hypothesis tracking for multiple target tracking. *Aerospace and Electronic Systems Magazine*, 2004.

[19] Ronald P. S. Mahler. *Statistical multisource-multitarget information fusion*. Artech House, 2007.

[20] Ronald P. S. Mahler. *Advances in statistical multisource-multitarget information fusion*. Artech House, 2014.

[21] Ba-Ngu Vo and Wing-Kin Ma. The Gaussian mixture probability hypothesis density filter. *Transactions on Signal Processing*, 2006.

[22] Kusha Panta, Daniel E. Clark, and Ba-Ngu Vo. Data association and track management for the Gaussian mixture probability hypothesis density filter. *Transactions on Aerospace and Electronic Systems*, 2009.

[23] Milos Vasic and Alcherio Martinoli. A collaborative sensor fusion algorithm for multi-object tracking using a Gaussian mixture probability hypothesis density filter. In *International Conference on Intelligent Transportation Systems (ITSC)*, 2015.

[24] Ronald P. S. Mahler. Optimal/robust distributed data fusion: A unified approach. In *Proceedings Volume 4052, Signal Processing, Sensor Fusion, and Target Recognition IX*, 2000.