Towards Large Scale Urban Traffic Reference Data: Smart Infrastructure in the Test Area Autonomous Driving Baden-Württemberg

Tobias Fleck*, Karam Daaboul*, Michael Weber*, Philip Schörner*
Marek Wehmer*, Jens Doll*, Stefan Orf*, Nico Sussmann°
Christian Hubschneider*, Marc René Zofka*, Florian Kuhnt*, Ralf Kohlhaas*
Ingmar Baumgart*, Raoul Zöllner° and J. Marius Zöllner*

*FZI Research Center for Information Technology, Karlsruhe
°Heilbronn University of Applied Sciences, Heilbronn

Abstract. This paper presents the concept, realization and evaluation of a flexible and scalable setup for smart infrastructure at the example of the Test Area Autonomous Driving Baden-Württemberg.
In verification and validation of autonomous driving systems, there exists a gap between virtual validation and real road tests: Simulation provides an easy and efficient way to assess a system's performance under a variety of environmental constraints, but is restricted to model assumptions and scenarios, which might ignore important aspects. Whereas expensive real road tests promise an unexpected environment for statistical evaluation of traffic scenarios, but lack of observability. Our setup for smart infrastructure is supposed to close the gap by tackling this issue by observing and providing reference data of traffic scenarios for application in different testing and evaluation settings.
We present the approach of implementing a distributed intelligent infrastructure capable of handling traffic light states, road topology and especially information about locally observed traffic participants. The data is provided online via Vehicle-to-X (V2X) communication for live testing and sensor range extension as well as offline via a backend for high-precision analysis and application of machine learning techniques. To obtain information about traffic participants, a camera based object tracking was realised. To cope with the high amount of information to be transmitted via V2X and to use the available bandwidth optimally, the standard for broadcasting vehicle information is modified by applying a form of data compression through prioritization.
The setup is initially evaluated at a large intersection in Karlsruhe, Germany.

1 Introduction

In order to pave the way for autonomous vehicles in different application fields such as individual or public transport and logistics, public test beds equipped with intelligent infrastructure are necessary to provide reference data and assist in test cases. The observed traffic situations in terms of realistic pedestrian or vehicle behavior, weather conditions, etc. then supports the development phase under different aspects. These test

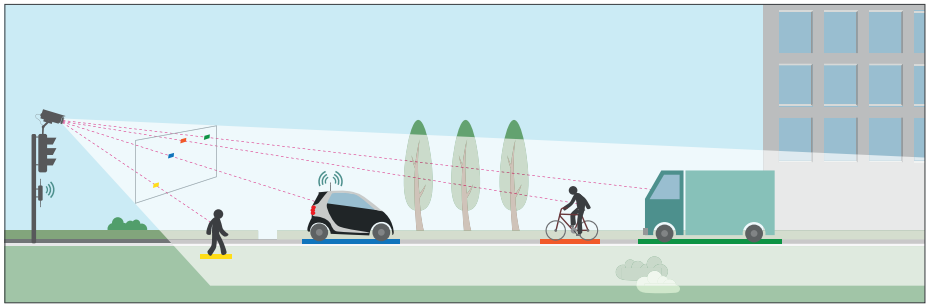


Fig. 1: Simplified overview of the smart infrastructure: Traffic participants are detected and tracked over space and time. The information is processed as anonymous, individual objects and provided online via V2X communication and stored offline for further processing. Thereby, the provided information enables the performance evaluation of automated vehicles (blue) or can be sent via V2X as additional sensor input to enhance their functionality.

beds with intelligent infrastructure are supposed to close the gap between virtual and real testing approaches for the release of autonomous driving systems.

In the *Testfeld Autonomes Fahren Baden-Württemberg*[1] (German for *Test Area Autonomous Driving Baden-Württemberg*, short: *Test Area*) selected streets of different type and complexity are equipped with intelligent infrastructure. This infrastructure is used to observe, process and communicate the present traffic situation as figure 1 shows. This enables to support the development and testing process of autonomous vehicles in a threefold manner:

1. The generated environment model is applicable for **online and offline evaluation** of the automated vehicle's environment model, typically captured from a restricted system's sensor view.
2. The observed traffic situations can be used as data pool for **algorithm development**, e.g to cluster and identify situation aspects for subsequent evaluation and validation techniques, such as scenario mining, improving models and test runs in simulation or as training data for machine learning.
3. The smart infrastructure is able to **extend the sensor range of automated vehicles** by supporting them with additional online information about the current traffic state, e.g. providing object lists or traffic light states. With the help of this data, sensor occlusion of participating vehicles can be compensated or at least mitigated.

With these applications in mind, we propose a concept for cost-efficient large scale intelligent infrastructure using decentralized local road units with low cost sensors and a central backend. The approach is used in the realization of the *Test Area Autonomous Driving Baden-Württemberg* within the cities of Karlsruhe, Heilbronn and Bruchsal. An early evaluation is given using a complex intersection in Karlsruhe.

[1] For more information about the *Test Area Autonomous Driving Baden-Württemberg* see `https://taf-bw.de/en`

The remainder of this paper is structured as follows: In Sec. 2 we give a brief overview over related work, considering existing test areas and smart intersections. Sec. 3 covers the overall concept, followed by a detailed explanation of the different algorithmic components (Sec. 4). We present a preliminary evaluation of our concept in Sec. 5 and conclude in Sec. 6 with open and ensuing research questions.

2 Related Work

Several field tests were conducted within Europe focusing on testing *Cooperative Intelligent Transport System* (C-ITS) use cases and deployments on public roads between 2014 and 2017 [1]. The term C-ITS refers to connected and automated mobility, with a strong emphasis on interacting vehicles and road infrastructure. The project *PRE-DRIVE C2X* [2] between 2008 and 2010 aimed at the establishment of a common European architecture for general V2X communication for future field operational tests (FOTs). It was continued between 2011 and 2014 with the project *DRIVE C2X* for the assessment of the technology in concrete cooperative driving functions within the field operational tests. The simTD [3] project in Germany evaluated C-ITS use cases and applications on public roads in different scenarios, along with a general architecture for C-ITS services. The SCOOP@F project evaluates the use of V2X equipment on a large scale on public roads in France [4]. The European Union finances the C-ROADS platform with the goal of coordinating the C-ITS rollout across the EU and developing specifications to ensure interoperability. A wider C-ITS deployment in general is envisioned to begin in 2019 [5]. The communication infrastructure used in this work conforms to standardized C-ITS-Protocols as much as possible with the goal of integrating well with a future large scale C-ITS rollout strategy.

Our approach differentiates from structured and self-contained proving grounds and dedicated test tracks, such as *MCity Test Facility* [6] or the closed-cource *Toyota Research Institute Automated Vehicle Test Facility* [7], in which only controlled experiments are conducted. Test in public traffic are desirable in order to evaluate a vehicle under unforeseen phenomena, such as non-modelable traffic participant behavior. At the same time observations by reference systems are needed.

In former publications mainly standalone data from intersections has been labeled manually in order to develop and evaluate algorithms for recognition tasks, such as the data provided by [8] for the development of tracking algorithms [9]. But mostly common datasets on the base of abstracted environment data are missing, which enable deriving elaborate behavior models and scenarios for further analysis or application in simulation. An initial attempt has been made in the german joint research project KoPER, where an intersection in Aschaffenburg has been equipped with multiple camera and lidar sensors. A part of the published dataset [10] was annotated by hand, in order to provide a set of object labels. The extensive use of multiple high precision sensors is promising for gaining high precision data. Though, for the use case of a persistent and widely spread deployment on different intersections and places, the scalability is limited by the high expenses of such a setup.

Besides the mentioned efforts, in Germany there are, as of today 15 ongoing initiatives to provide so called digital test beds on motorways and in cities but also covering

cross-border test areas [11]. In Austria, the alliance *ALP.Lab GmbH* [12] by automotive supplier companies, AVL and Magna, and scientific partners, Joanneum Research, TU Graz and Virtual Vehicle, has developed a platform for testing and verifying the components and systems of automated driving in diverse and complex scenarios including facilities for data recording and processing along public roads and privat grounds as well as comprehensive virtual testing environments.

3 Concept

Since the Test Area Autonomous Driving Baden-Württemberg focuses on all three applications described in Sec. 1 (sensor extension, online/offline evaluation and algorithm development) several requirements have to be considered to develop an efficient overall system.

Our approach is depicted in Figure 2: For offline evaluation and longterm algorithm development a backend in a data center is used while local processing is used to deliver online information to the local automated vehicles. The local processing is implemented in local road units that span over areas such as a connecting street or an intersection. The local environment is observed by the local road units using object sensors. To comply with privacy regulations, data with personal information such as image

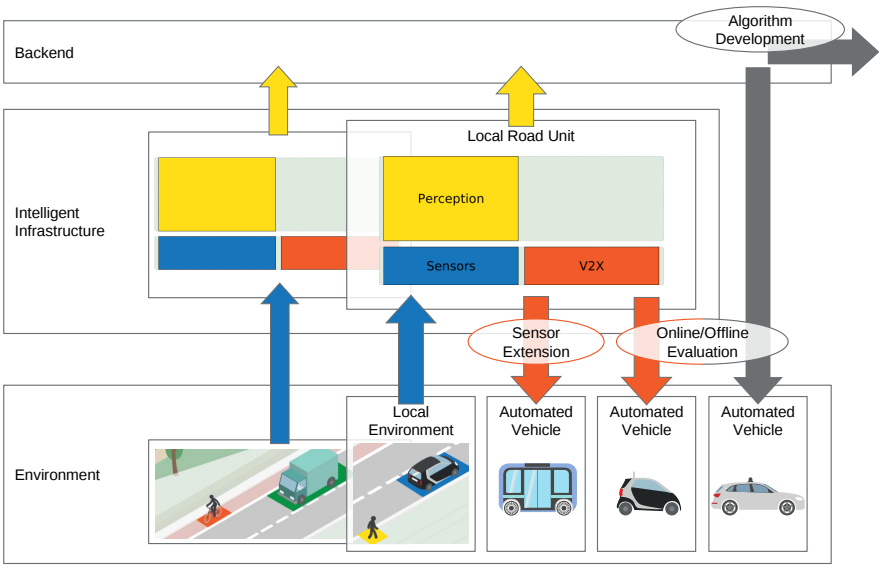


Fig. 2: Overview of the concept and applications in the Test Area Autonomous Driving Baden-Württemberg. Local road units perceive the environment and deliver processed data to the backend and local automated vehicles. The three applications sensor extension, online/offline evaluation and algorithm development (using recorded data as training data) are depicted.

data is anonymized as early as possible and not delivered to the backend nor communicated via V2X. Likewise, the anonymized data is also smaller and easier to deliver to the backend over a long distance. For a large scale deployment we focus on cheap sensors, such as 2D image cameras, but additional sensors such as radar or lidar can be integrated. Besides the observed objects, the local road units additionally have knowledge about the traffic light states and a local map of the road topology. This allows on the one hand to transmit a coherent, complete, processed view of the current local traffic scene to the backend, on the other hand local communication is possible without relying on the backend.

Thus, the key components of the approach are the local road units. They can be deployed as permanent installations or mobile solutions for temporary experiments. In this work we focus on permanent installations in urban areas. These are initially deployed in the Test Area Autonomous Driving Baden-Württemberg which will be extended with more infrastructure during the next years.

3.1 Local Intelligent Infrastructure

A visualization of the concept of the local road units is shown in Fig. 3. We see the units as small intelligent agents observing and communicating with their environment. Thus, we follow the typical perception-cognition-action approach [13] but with a focus on perception. This allows concepts and interfaces to be reused from automated driving

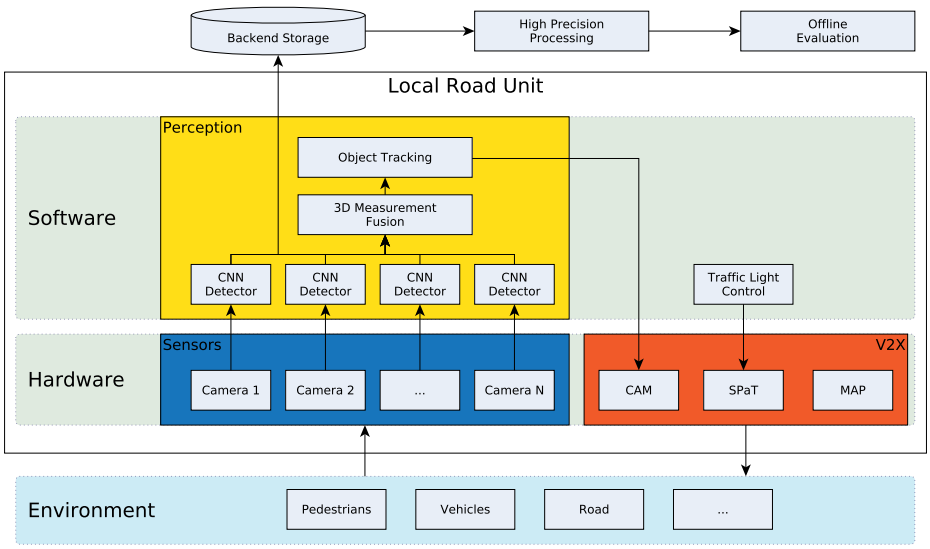


Fig. 3: Components inside the local road units including interfaces to the environment and the backend. The local road unit is divided into hardware components as sensors and V2X communication unit and software components for the processing of sensor data and the traffic light control.

implementations (compare [14]) including the utilization of a simulation framework during development [15].

In order to perceive the traffic situation, multiple optical camera sensors are attached to large poles with overlapping fields of view. The multi-camera setup has to be calibrated to determine the exact intrinsic and extrinsic parameters for retrieving correct geometric information from the images, see Sec. 4.1. The undistorted images are used as input for Convolutional Neural Networks (CNN) to detect and classify vehicles, pedestrians and cyclists in the 2D image space, see Sec. 4.2. For high precision offline processing the output of the single CNN detectors can be transmitted to the backend storage. This allows maximum flexibility in processing while not offending against privacy regulations and keeping a low amount of data (compared to saving the actual raw sensor data). In order to obtain 3D objects in world coordinates for online usage, the classification hypotheses are then fused to single 3D measurements (Sec. 4.3) and tracked over time by a Bayesian filter associating a series of measurements with potential vehicles, yielding spatio-temporal tracks for perceived traffic participants (Sec. 4.4). Recognized traffic participants are represented as a list of abstract objects with a predefined set of features. It is transformed to a standards-compliant set of messages and wirelessly broadcasted via Vehicle-to-X communication. This enables users of the Test Area to receive the traffic state in-place and at the same time represents a C-ITS pilot test site for all vehicles. We describe the communication in detail in Sec. 4.5.

3.2 Perception Problem Factorization

The overall estimation problem of the perception module can be formulated as estimating the state of all objects X given all sensor measurements with uncertainties M and background knowledge θ (e.g. intrinsic/extrinsic calibration, learned detection model parameters), see Fig. 4:

$$X \sim P(X|M,\theta)$$

This problem is too complex to be modeled in its entirety. Thus we factorize the problem into subproblems (Fig. 5) that can either be parametrized by expert knowledge or learned using machine learning techniques, as we explained in our previous work [16]. Multi-Sensor Multi-Target Tracking (MTT) is a research topic for decades with various approaches. Recent work [17] tries to formalize all estimation approaches including the

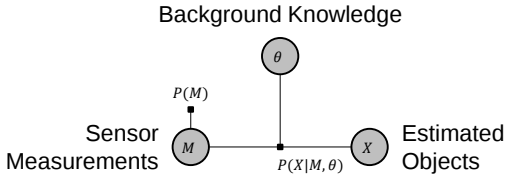


Fig. 4: Object estimation problem statement: The set of objects X has to be estimated, given all sensor measurements M and background knowledge θ.

big fields of vector-type MTT methods (e.g. Kalman Filter, Interacting Multiple Model Filter) and set-type MTT methods (e.g. PHD Filter). This formalization can be recognized in Fig. 5 with the difference that we include the whole preprocessing chain and use an early sensor fusion.

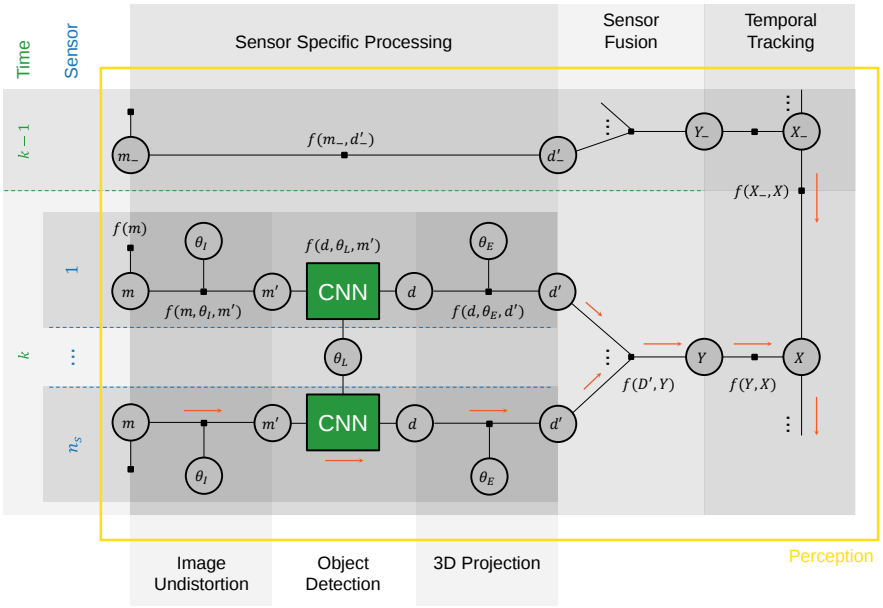


Fig. 5: Our approach of factorizing the perception problem $P(X|M,\theta)$ into subproblems: Per timestep the detections $D' = \{d'^1, \ldots, d'^{n_s}\}$ of a sensor specific preprocessing are fused to 3D measurements Y. A Bayesian filter tracks object estimates X over time using the Markov assumption. The sensor specific processing consists of image undistortion using the sensor-specific intrinsic parameters, CNN-based 2D object detection and conversion to a common 3D reference coordinate system using the extrinsic parameters and assuming a flat ground plane. The notation is inspired by [17] and [18].

4 Implementation

4.1 Camera Calibration

In order to register detections of all cameras in a common reference coordinate system, a proper camera calibration is needed. This includes estimating the intrinsics of all cameras respectively and to estimate the extrinsic calibration of all cameras relative to a reference coordinate system all objects and vehicles should be tracked in. The camera calibration is assumed to be static over time and thus is only performed once a-priori to the processing pipeline.

Due to manual focus adjustments all cameras are calibrated intrinsically after mounting. This is commonly done by using calibration patterns with geometric forms that can easily be detected by computer vision algorithms. The intrinsic parameters are calculated by minimizing the reprojection error iteratively with the Levenberg-Marquardt algorithm [19]. Extrinsic calibration can also be achieved by using calibration patterns. This would require a single pattern visible from all cameras or multiple patterns which themselves must be extrinsically calibrated. Both cases are infeasible for a large scale intersection as they either require temporarily disturbing traffic or high-cost equipment. To overcome this problem, we use the a-priori high precision maps of the test area for reference point localization. Together with the intrinsic parameters the thereby known real-world positions of features in camera images can be used to estimate the extrinsic calibration for each camera. Dominant features like lane markings or poles of traffic signs that are visible in multiple camera images and are recorded in the a-priori map are manually selected to be used in the calibration process (see Fig. 6). The exact position of these features in the reference coordinate system is known through the a-priori map and also in the image coordinate system. The pose of the camera is then iteratively estimated by minimizing the sum of squared distances between image points and projections of the real-world points, again using the Levenberg-Marquardt algorithm. The calibration then allows to calculate the real-world positions of objects in the camera image.

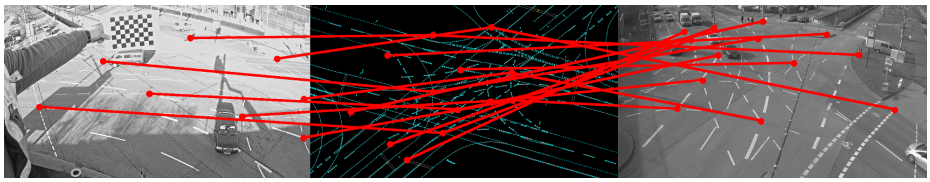


Fig. 6: Manual feature selection in camera images (left and right) and correspondence with a reference map (middle) for extrinsic calibration. Additionally, on the left image an calibration pattern for intrinsic camera calibration is visible.

4.2 Object Detection

One of the key tasks in the implemented processing pipeline (Fig. 5) is detecting and classifying vehicles in undistorted monocular camera images m'. The results d are two-dimensional object surrounding bounding boxes in image coordinates with a probability distribution over predefined object classes. In recent years, *Convolutional Neural Networks (CNNs)* became state of the art in solving such object detection tasks. There exist two different groups of algorithms for 2D object detection. Proposal based approaches like Region-CNN [20] and its successors Fast R-CNN [21] and Faster R-CNN [22] apply a region proposal algorithm on the image. This proposes candidates of objects which are feed into a classifier in a second step. Different to this pipelined approach, the other group of algorithms directly detects and classifies objects within one single network in a single step. Examples for this group are Overfeat [23], YOLO [24] and SSD. These algorithms usually do classification and object position estimation for each region within an image. The position estimations for objects are typically filtered by confidence values and afterwards clustered or a non-maximum suppression is applied. We approach the detection problem by using an implementation of Mask R-CNN [25], an artificial convolutional neural network which is state-of-the-art in image detection, trained on publicly available datasets ([26],[27]). The output d includes a probabilistic estimation of the classification but no uncertainty measures on the position estimates.

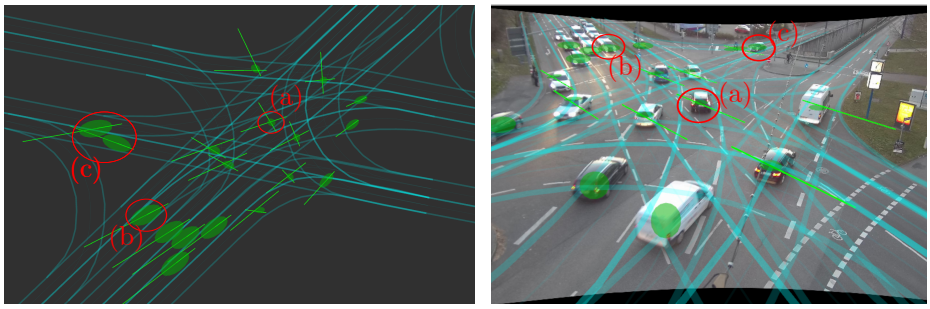
4.3 Multi Sensor Association and Fusion

Given detected classified bounding boxes d in image coordinates from the object detection module, an association and fusion module generates appropriate position measurements Y in a predefined 3D reference coordinate system (Fig. 5).

For each camera the detected objects in image coordinates are projected to a parametric model of the ground plane. The model can be a simple hyperplane in most cases or a more precise elevation model if necessary. The projection is achieved by intersecting a parametric line from the camera origin through the pixels of the referring image detection, with the parametric ground model, leading to a 3D position measurement d' (Fig. 1). Thus, every camera leads to a set of position measurements relative to the reference frame the cameras are calibrated against. To take into account the measurement uncertainties (that are not delivered by the CNN detector), zero mean Gaussian noise is assumed on the detections d. The projection from detections to the ground plane model distorts the distribution. As a first approximation the projected measurements d' are assumed to have again Gaussian distributed noise with the following properties: the eigenvectors of the covariance matrices are aligned with the direction of projection and moreover uncertainty on the position measurement is high in the projection direction, but low on the perpendicular direction to it. To deal with calibration errors, we add additive zero mean Gaussian noise on each measurement that increases with the distance of the measurement to the camera.

In a second step, measurements from each set $d'^1, \ldots, d'^{n_s}$ are mutually associated by computing a minimum cost matching between the computed sets using the euclidean distance of the position measurements as a cost function. Once measurements from different sets have been associated, they are fused into one measurement considering

their individual noise model. To prevent the fusion of measurements that are far apart, a common gating step is included before the fusion. Figure 7 shows an example scene with an illustration of the resulting measurement covariances of Y.



(i) Generated 3D measurements from a given detection in the image frame.

(ii) The corresponding monocular camera image with 3D measurements projected back into the image.

Fig. 7: Example data from the measurement fusion process: Green line segments represent parametric lines that are intersected with a ground plane model. Green ellipses correspond to covariance matrices of the relating measurements. The red circles mark three representative cases the fusion process might lead to:
(a) The measurements of two cameras were successfully fused.
(b) The object is only recognized from one camera, therefore there is no fusion of measurements.
(c) The two cameras have recognized the object, but the resulting intersections with the ground plane are too far apart and therefore no fusion takes place.

4.4 Temporal Association and Multi-Target Tracking

Once the vehicles have been located in the reference frame, the next task is to maintain their identity and determine their individual trajectories. To solve these problems, a filter for tracking multiple objects $X = \{x^1, \ldots, x^{n_t}\}$ and an algorithm for assigning measurements $Y = \{y^1, \ldots, y^{n_s}\}$ to tracks are required. The state vector x^i of each tracked vehicle contains the position of the object, its linear and angular velocity and its linear acceleration and the measurement vector y^i describes the position of a possible vehicle in 3D space. The estimation and prediction of the state vector is done using an *interacting multiple model filter (IMM)*[28] with three prediction models: *constant turn-rate and velocity (CTRV)*[29], *constant acceleration (CA)*[28] and *constant location (CL)*[28]. This way, all common movement types for vehicles at urban roads and intersections are incorporated neatly into the estimation process.

Association of measurements to tracks is done using a *global nearest neighbor (GNN)* approach. Herefore, an optimal bipartite matching between current track hypotheses and received measurements is computed using the hungarian algorithm [30] with a distance measure based on the covariance noise model of the estimated tracks

X and the measurements Y. A gating procedure completes the assignment and supports track management.

A track can have different management states: *valid*, *invalid* and *potential track*. Each possible track starts as a potential track and when the uncertainty on its estimated state vector deceeds a certain threshold, the track is set to be valid. When a track or track hypothesis is not updated with measurements, for instance when the track leaves the field of view, the uncertainty of the state estimate increases until a parametrized threshold is exceeded then the state is set to invalid.

The result is a list of tracked objects per time step and can be transmitted to the Vehicle-to-X communication module.

4.5 Vehicle-To-X Communication

In addition to collecting tracking information in the backend for offline analysis, an online low-latency transmission of the current traffic situation is enabled. Thus making all tracked vehicles available for autonomous vehicles in the local environment of the intelligent infrastructure. Instead of using a proprietary protocol existing V2X standards are leveraged to communicate the intersection state to nearby vehicles. By adhering to the European ITS-G5 specifications we reduce the additional setup required by users of the Test Area. An overview of the used architecture is given in Fig. 8.

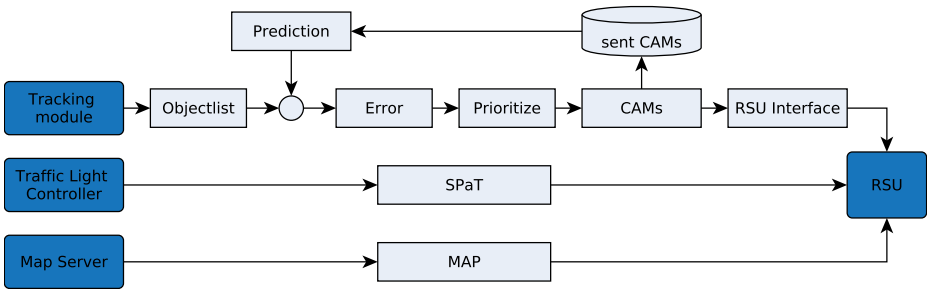


Fig. 8: Overview of Vehicle-to-X communication: Sending information about tracked vehicles via CAMs using a RSU-bridge to connect our V2X pipeline to the RSU. Broadcasting traffic lights states in the form of SPaTs and providing the intersection topology through MAP messages is directly done by the traffic light controller. A single road side unit is used as common transmitter.

Sending V2X messages for other vehicles is not strictly covered by existing V2X standards, but since very few consumer vehicles are equipped to send or receive such V2X messages, the benefits for testing within a more complete V2X setup by emulating V2X capabilities of other traffic participants outweighs the interference with the standard. Also, it is expected that more and more consumer vehicles will possess V2X capabilities in the future, which future version of the Vehicle-to-X pipeline need to consider. One possible solution would be the detection of vehicles actively sending V2X messages and their removal from the internal transmission list.

The European Telecommunications Standards Institute (ETSI) specifies a protocol stack along with a message format for inter-vehicle information broadcasts containing the position, heading, movement state and general information about the sending vehicle [31]. These broadcasts are named *cooperative awareness message* (CAM) and are sent repeatedly by vehicles to inform neighbors about their presence.

Our implementation estimates these features as part of the tracking algorithm and generates CAM messages for every observed vehicle. A deployed V2X roadside unit (RSU) is used to physically send the generated CAMs. In practice our pipeline is limited to a message rate of 400 to 600 CAMs per second. While this is enough for the currently used setup, we need to be prepared for future extensions and requirements. To avoid high latencies and dropped messages in the future, an additional data compression step is applied that uses prioritization and knowledge about the development of the scene to minimize the required amount of messages, which is described at the end of this section.

Beside of the transmission of the traffic participant's behaviour, the RSU also transmits the intersection's topology as well as the current traffic lights' states as depicted in the lower part of Fig. 8. For this the ETSI TS 19 091 / ISO TC 204 message standards for signal phase and timing (SPaT) and intersection geometry and topology (MAP) are used. In conjunction, this enables the perception of the intersection's lanes geometry and transitions between them as well as which are currently allowed to be drivable.

Using the allocated bandwidth for CAM messages optimally becomes important if the number of vehicles increases. The main idea of the applied compression is to reduce the transmission of CAM messages based on the estimated error the prediction on the receiving side will observe. For this the sending module keeps track of all CAMs that were already sent and uses the same prediction model a theoretical receiver would use. Assuming no message loss occurs during transmission the error can be estimated by predicting the last sent message and comparing its outcome with the newly available measurement. Comparing the estimated error over all tracked vehicles allows for a message prioritization scheme which minimizes the overall observed error when only a fixed number of entries in the transmission list are actually send over the air. This idea is similar to the decentralized frequency management for vehicle ITS stations mechanism specified in ETSI EN 302 637-2, which controls the CAM frequency based on the vehicle dynamics, but using the knowledge about the complete scenario. An overview of the process is shown in the upper part of Fig. 8.

Information about vehicles with a large derivation from the predicted state are therefore sent with a higher frequency than others. Since the prediction on the receiving side is in general not known, the aim is to estimate the upper bound of the prediction error This is done by using a basic constant velocity predictor, although acceleration values are available within the sent messages. The usage of advanced prediction methods or higher order predictors on the receiving side will invalidate the assumption about the prediction error on the sending side and therefore result in a prioritization scheme that isn't optimal any more, however the prediction error will be in general lower than the sending side estimates and thus the overall accuracy will improve. Using those advanced prediction methods on both sides would further increase the effectiveness of proposed concept on data compression, which is evaluated in section 5.

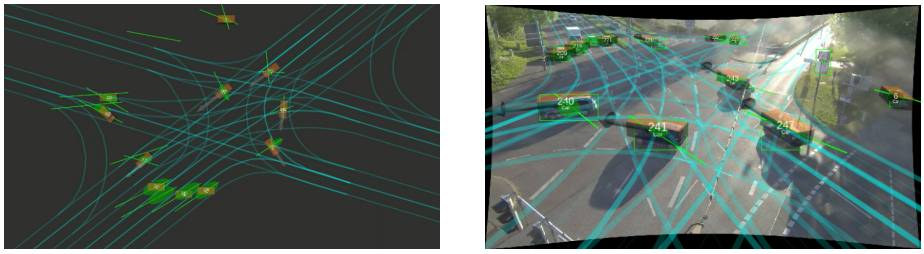


Fig. 9: Example tracks from dataset 2 containing strong backlight towards the cameras. Orange boxes visualize tracked objects and green boxes refer to object detections in the input image.

5 Evaluation

In the following we will give a short preliminary evaluation of the proposed concept on the object recognition and V2X communication based on real data from an exemplary local road unit located at a large intersection in Karlsruhe.

5.1 Hardware Setup

The exemplary local road unit consists of two cameras, a server for the image processing and tracking software and a *roadside unit (RSU)* for V2X communication. All hardware components are connected via gigabit ethernet switches. In order to transmit the current traffic light status, the RSU is additionally connected to the *traffic light controller (TLC)*. For this we use the second networking interface of the RSU. Except for the (unidirectional) communication between the TLC and the RSU, our processing infrastructure is completely isolated from potentially safety-critical intersection hardware units.

The hardware located in the intersection itself is connected via LAN to a data center for monitoring, administration and data storage purposes, and is not directly accessable from public internet.

5.2 Object Recognition

The complete pipeline from object detection to tracked 3D objects is evaluated end-to-end on two datasets from the equipped example intersection. The first dataset was recorded under good weather conditions, while the second dataset contains heavy backlight towards the cameras (Fig 9). Both datasets are compared to the processed object estimations manually using a projection from the estimated 3D boxes back to the input images. Our analysis takes different cases into account.

The *True Positive Case (TP)* is the case in which the vehicle is detected by at least one of the cameras and the tracker can follow this vehicle from the point at which it enters the intersection to the point where it leaves the field of view. The case in which an object other than a vehicle such as a traffic sign or a pedestrian is detected and

tracked as a vehicle is the *False Positive Case (FP)*. If the camera is not able to detect an existing vehicle or the tracker does not track it at all, we have a *False Negative Case (FN)*. The case when the tracker detected a vehicle but was not able to keep its track until the vehicle leaves the field of view, is listed as *Lost Track Id*. Table 1 summarizes the gained results.

	TP	FP	FN	
Setting	Detected Vehicle	Detection w/o Vehicle	Undetected Vehicle	Lost Track Id
Dataset 1	65	0	1	5
Dataset 2	88	4	1	11

Table 1: Evaluation results of the implemented object tracking pipeline. Dataset 1 contains good lighting conditions, while heavy backlight against the cameras is present in Dataset 2.

One of the reasons that causes the loss of tracks is that the gained measurements have a low frequency (about 5–6hz), so several prediction steps have to be made between two measurements, causing difficulties with the association of measurements to tracks. The high number of False Positives in the second recording is caused by structural false detections of the CNN due to the high backlight towards the camera. This behavior can be mitigated by placing more cameras with different angles into the intersection and by retraining the detection algorithm for such lighting conditions with appropriate data.

5.3 Vehicle-to-X

Correctness We confirmed the correctness of the received CAMs visually by deriving oriented bounding boxes and their position from the messages. Afterwards the bounding boxes are reprojected into the camera images and verified that they align with visible vehicles.

Performance The empirically estimated transmission rate of 400 to 600 CAMs per second allows to emulate V2X messages for a maximum of 60 vehicles, assuming a maximum allowed message rate of 10 Hz per vehicle. But to avoid latency during the transmission, a bandwidth of 400 CAMs per second is chosen.

Having 12 ingress lanes, the presented intersection is rather large and carries a lot of traffic at peak times. We estimate from manual observations that a number of 50 vehicles is no exception. However the current perception pipeline is only tracking up to 25 vehicles at the same time, as currently only vehicles on the inner intersection area are recognized. Since this number will increase in the future and to avoid limitations of the service quality, the goal of at least 100 objects to handle at the same time was set.

To utilize the allocated bandwidth optimally, the previously introduced concept for data compression is employed, which takes advantage of the prediction capabilities on the receiver side to reduce the overall error. The maximum improvement depends on

the actual degree of unpredictability of other traffic participants. According to the ETSI standard the following frequencies are allowed: For the worst case a 10 Hz rate is needed to cover unpredictable vehicles and a (best case) 1 Hz rate to cover the predictable vehicles as the allowed simplification. Therefore the number of tracked vehicles per second r_v, as the sum of number of predictable vehicles $n_{predictable}$ and number of unpredictable vehicles $n_{unpredictable}$ per second, is as follows

$$r_v = \frac{r_{messages}}{q_m} = \frac{400}{q_m} \tag{1}$$

with $r_{messages}$ as the desired number of messages per seconds and q_m being the average number of messages per vehicle

$$q_m = \frac{10 * n_{unpredictable} + n_{predictable}}{n_{unpredictable} + n_{predictable}} \tag{2}$$

$$= 1 + 9 * \frac{n_{unpredictable}}{n_{unpredictable} + n_{predictable}} = 1 + 9 * q_u. \tag{3}$$

We define q_u in equation 3 as the relative amount of unpredictable vehicles. To properly handle a rate of $r_v = 100$ tracked vehicles per second crossing the intersection with $r_{messages} = 400$ CAMs per second, a message quotient of $q_m = {}^{400}/_{100} = 4$ is needed, meaning the ratio of unpredictable vehicles $q_u = {}^{(q_m-1)}/_{9}$ must be lower than 0.33. As q_u depends only on traffic characteristic, it can be measured with empirical data and we assume it is independent of the actual amount of vehicles.

For the empirical measurement of q_u a distance error of 0.5 meters is set as threshold value. Every 250 ms, the difference of the predictor output and the incoming measurement is measured for each tracked vehicle. If the error is larger than the threshold, the vehicle is counted as unpredictable, else it is counted as predictable for this time step. At each time step, we calculate q_u and skip time steps with no tracked vehicles. In

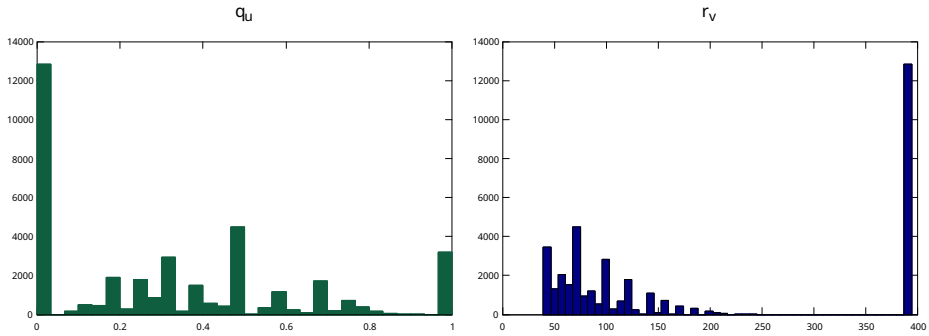


Fig. 10: Histogram of values of q_u and r_v: The right diagram shows a histogram of q_u with a constant velocity predictor. Values of $q_u = 0$ are mostly situations, where no or one standing vehicle is visible. Values $q_u = 1$ are typically caused by situations with a few vehicles following a turn lane. The left diagram shows the corresponding histogram of r_v values, where r_v is the theoretically limit of vehicles supported by given bandwidth of 400 CAMs per second.

recordings of 3 hours an average q_u value of 0.3281 could be seen, which is very close to our target value of 0.33. The value was higher than expected. But since the focus of the sensor setup is designed to be at the center of the intersection, the value still is plausible. In this case, the total number of vehicles was never higher than 25. Still we can not completely rely on the average, as there may be periods where the q_u is higher for longer times, which results in high message latencies during these periods if the number of vehicles is high enough. Figure 10 shows histograms of the q_u and r_v values. While we are well above the 100 vehicles per second mark most of the time, we still have a lot of time steps below this value. Widening the angle of view would bring more standing vehicles into the field of view and therefore yield a lower q_u value, as would using a more complex predictor. Apart from this, the messaging rate using the RSU might need to be improved to scale the system up for future challenges.

6 Conclusion

We presented the ideas, concepts and algorithms for a generic, flexible and scalable setup for intelligent infrastructure that can support the development of automated vehicles. We explained it using the application in the Test Area Autonomous Driving Baden-Württemberg and evaluated the object recognition and Vehicle-to-X (V2X) modifications in an exemplary intersection in Karlsruhe. The intelligent infrastructure consists of cameras used as perception sensors and accompanying V2X communication and is able to provide online as well as offline data from the environment, containing dynamic objects and states of traffic lights. Dynamic objects are detected within multiple camera streams using CNN detectors, associated with the road plane and then temporally tracked using an IMM filter.

Recorded data can be used offline as reference for perception system of automated vehicles currently traversing the intersection on the one hand. This includes evaluating the sensor setup of a vehicle and identifying dead spots within the setup while additionally checking the correctness and consistency of sensors and the ensuing perception and cognition algorithms of vehicles under test. The data could also be used to recreate and modify special, real-world situations in simulation environments. On the other hand, data can be broadcasted via V2X to tackle the occlusion problem for automated vehicles that could lead to potentially dangerous situations by providing additional environment information to and extending the sensor horizon for vehicles, potentially enabling vehicles with smaller sensor setups.

Although a first version of the concept has been implemented and deployed successfully, the concept and individual parts in the processing chain will be further extended and improved. As a next step, dynamic object tracking will be extended to cyclists and pedestrians. Furthermore, increasing the observed area within intersections as well as covering the roads leading into and connecting multiple intersections will provide much more valuable information about detailed traffic movement and is a next step in our rollout.

Another current shortcoming is the manual calibration process to include new cameras. An online recalibration and automated calibration of the cameras will guarantee

a persistent quality level over time and an improved elevation model of covered road segments can accommodate non-planar road geometry.

Acknowledgement

This work was done within the project ”Digitales Testfeld BW für automatisiertes und vernetztes Fahren”, referred to as ”Testfeld Autonomes Fahren Baden-Württemberg”, funded by the Ministry of Transport Baden-Württemberg.

Under the direction of the FZI Research Center for Information Technology, a consortium of the City of Karlsruhe, the Karlsruhe Institute of Technology, Karlsruhe University of Applied Sciences, Heilbronn University of Applied Sciences, the Fraunhofer Institute for Optronics, System Technology and Image Evaluation IOSB and the City of Bruchsal and other associate partners is implementing the development of the Test Area.

References

1. European Commission, “Cooperative, connected and automated mobility (C-ITS),” https://ec.europa.eu/transport/themes/its/c-its_en, accessed: 2018-05-15.
2. A. Tomatis, M. Miche, F. Haeusler, M. Lenardi, T. M. Bohnert, and I. Radusch, “A test architecture for V-2-X cooperative systems field operational tests,” in *International Conference on Intelligent Transport Systems Telecommunications (ITST)*, 2009.
3. D. Hübner and G. Riegelhuth, “A new system architecture for cooperative traffic centres - the simTD field trial,” *19th ITS World Congress*, 2012.
4. H. Aniss, “Overview of an ITS Project: SCOOP@F,” in *Communication Technologies for Vehicles*. Springer International Publishing, 2016.
5. K. Sjoberg, P. Andres, T. Buburuzan, and A. Brakemeier, “Cooperative Intelligent Transport Systems in Europe: Current Deployment Status and Outlook,” *IEEE Vehicular Technology Magazine*, 2017.
6. “MCity Headquarters: MCity Test Facility,” https://mcity.umich.edu/our-work/mcity-test-facility/, accessed: 2018-05-20.
7. “Toyota Research Institute: Opening in October - Toyota Research Institute Automated Vehicle Test Facility,” http://www.tri.global/news/opening-in-october-toyota-research-institute-auto-2018-5-3, accessed: 2018-05-20.
8. J. P. Jodoin, G. A. Bilodeau, and N. Saunier, “Urban Tracker webpage,” https://www.jpjodoin.com/urbantracker/dataset.html, accessed: 2018-05-20.
9. J.-P. Jodoin, G.-A. Bilodeau, and N. Saunier, “Urban Tracker: Multiple Object Tracking in Urban Mixed Traffic,” in *Winter Conference on Applications of Computer Vision (WACV)*, 2014.
10. E. Strigel, D. Meissner, F. Seeliger, B. Wilking, and K. Dietmayer, “The Ko-PER Intersection Laserscanner and Video Dataset,” in *International Conference on Intelligent Transportation Systems (ITSC)*, 2014.
11. “Federal Ministry of Transport and Digital Infrastructure: Digital Test Beds,” http://www.bmvi.de/EN/Topics/Digital-Matters/Digital-Test-Beds/digital-test-beds.html, accessed: 2018-05-15.
12. “ALP.Lab GmbH (Austrian Light Vehicle Proving Region for Automated Driving),” http://www.alp-lab.at/, accessed: 2018-05-15.

13. S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2016.
14. F. Kuhnt, M. Pfeiffer, P. Zimmer, D. Zimmerer, J. M. Gomer, V. Kaiser, R. Kohlhaas, and J. M. Zöllner, "Robust environment perception for the Audi Autonomous Driving Cup," in *International Conference on Intelligent Transportation Systems (ITSC)*, 2016.
15. M. R. Zofka, F. Kuhnt, R. Kohlhaas, and J. M. Zollner, "Simulation framework for the development of autonomous small scale vehicles," *International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR)*, 2016.
16. C. Hubschneider, J. Doll, M. Weber, S. Klemm, F. Kuhnt, and J. M. Zöllner, "Integrating End-to-End Learned Steering into Probabilistic Autonomous Driving," in *International Conference on Intelligent Transportation Systems (ITSC)*, 2017.
17. F. Meyer, T. Kropfreiter, J. L. Williams, R. A. Lau, F. Hlawatsch, P. Braca, and M. Z. Win, "Message Passing Algorithms for Scalable Multitarget Tracking," in *Proceedings of the IEEE*, vol. 106, no. 2, 2018.
18. T. Heidenreich, J. Spehr, and C. Stiller, "LaneSLAM - Simultaneous Pose and Lane Estimation Using Maps With Lane-Level Accuracy," in *International Conference on Intelligent Transportation Systems (ITSC)*, 2015.
19. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov 2000.
20. R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Computer Vision and Pattern Recognition (CVPR)*, 2014.
21. R. Girshick, "Fast R-CNN," in *International Conference on Computer Vision (ICCV)*, 2015.
22. S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Neural Information Processing Systems (NIPS)*, 2017.
23. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," 2014.
24. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Computer Vision and Pattern Recognition*, 2016.
25. K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *International Conference on Computer Vision (ICCV)*, 2017.
26. T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision (ECCV)*, 2014.
27. M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Computer Vision and Pattern Recognition (CVPR)*, 2016.
28. C. Barrios and Y. Motai, *Predicting Vehicle Trajectory*. CRC Press, 2017.
29. Y. Bar-Shalom, P. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, 2011.
30. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.
31. "EN 302 637-2 V1.3.2; Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service," ETSI, Sophia Antipolis Cedex - FRANCE, Standard, Nov. 2014.