



Information and Communication Technologies

Collaborative Project

“V-Charge”

Autonomous Valet Parking and Charging for e-Mobility

Grant Agreement Number 269916



Deliverable Number	D6.9
Deliverable Name	Final Test Report
Dissemination Level	Non public
WP Number	6
Date of Preparation	26.10.2015
Date of Submission	30.10.2015
Editor/Beneficiary	Ulrich Schwesinger / ETHZ

Executive Summary

This deliverable summarizes the performance of the final V-Charge platform in the different scenarios specified in Deliverable 6.1. The data to evaluate the system in these scenarios was collected in the run-up to the final demonstrations. Statistics presented in this document are limited to the time period of three month (April – June 2015) under which the performance of the system was close to its final state. In total a number of 509 trials for all the scenarios were conducted.

In contrast to the individual workpackage deliverables targeting the evaluation of subcomponents of the system, the goal of this deliverable is to summarize the performance of the system as a whole. In case of decreased performance, this document highlights sources of problems and suggests solutions in order to facilitate follow-up work. The status of each scenario is classified as either “achieved”, “partially achieved” or “not tackled”. The label “achieved” classifies the performance of the system in the scenario as reliable and appealing. “Partially achieved” indicates that all functionality was implemented and deployed in the car, yet the performance was not fully satisfactory. “Not tackled” refers to scenarios which were not tested due to missing functionality in the software chain.

It is recommended to read the accompanying deliverables 2.4, 3.5, 4.4 and 5.3 before this document.

Executive Summary	2
Scenario 1: Driver arrives at the drop-off zone and activates the system	4
Scenario specification	4
Observable actions of the system.....	4
Necessary functionality.....	4
Results.....	4
Scenario 2: Driver summons the car and deactivates the system	7
Scenario specification	7
Observable actions of the system.....	7
Necessary functionality.....	7
Results.....	8
Scenario 3: Robot drives through parking area.....	8
Scenario specification	9
Observable actions of the system.....	9
Necessary functionality.....	9
Results.....	9
Scenario 4: Parking into and out of a parking spot or charging station	10
Scenario specification:.....	10
Observable actions of the system:	10
Necessary functionality:	10
Results.....	11
Scenario 5: Robot adjusts the route due to an obstacle blocking the road	11

Scenario specification	11
Observable actions of the system.....	12
Necessary functionality.....	12
Results.....	12
Scenario 6: Robot passes an obstacle at the side of the lane	12
Scenario specification	13
Observable actions of the system.....	13
Necessary functionality.....	13
Results.....	13
Scenario 7: Robot follows a vehicle in the front	14
Scenario specification:	14
Observable actions of the system:	14
Necessary functionality:	14
Results.....	15
Scenario 8: Robot passes an oncoming vehicle.....	16
Scenario specification	16
Observable actions of the system:	16
Necessary functionality:	16
Results.....	16
Scenario 9: Robot lets other vehicle pull out of a parking spot	17
Scenario specification	18
Observable actions of the system.....	18
Necessary functionality.....	18
Results.....	18
Scenario 10: Pedestrian is crossing the road in front of the car	18
Scenario specification	19
Observable actions of the system.....	19
Necessary functionality.....	19
Results.....	19
Scenario 11: Robot passes a pedestrian walking at the side of the road	20
Scenario specification	20
Observable actions of the system.....	20
Necessary functionality.....	20
Results.....	20
Scenario 12: Robot pulls out of parking spot in presence of traffic.....	21
Scenario specification	21

Observable actions of the system.....	22
Necessary functionality.....	22
Results.....	22
Scenario 13: Robot approaches and passes an intersection.....	22
Scenario specification	23
Observable actions of the system:	23
Necessary functionality:	23
Results.....	23
Scenario 14: Robot reacts to an emergency signal	24
Scenario specification:.....	25
Observable actions of the system:	25
Necessary functionality:	25
Results.....	25
Summary and Conclusion	26

Scenario 1: Driver arrives at the drop-off zone and activates the system

Maturity level: basic

Status: **ACHIEVED**

Success rate: 91% (43/47)

Main error source(s): Wifi/Mobile Link connectivity problems

Scenario specification

- driver arrives at the drop-off zone
- driver activates the system
- driver leaves the car and locks the doors

Observable actions of the system

- System activation via smartphone

Necessary functionality

- HMI: driver to smartphone
- vehicle to server communication
- mission start, activation of the autonomous mode
- global path planning

Results

This scenario relates to the vehicle-to-infrastructure components developed in workpackage 5. For a successful scenario the smartphone application and vehicle-to-server communication have to be in place.

Figure 1 gives a high-level overview of the general communication components involved in this task.

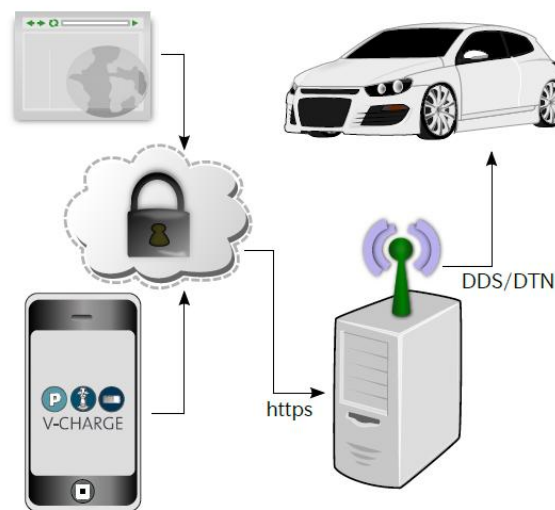


Figure 1 High-level overview of communication components

The software architecture of the V-Charge backend service ParkingManager (PM) has been described in Deliverable 5.2 (D5.2). We refer the reader to D5.2 for a detailed description. Figure 2 shows a more detailed view of the V2X communications architecture. DDS Interface Definition Languages (IDLs) have been described in D5.3.

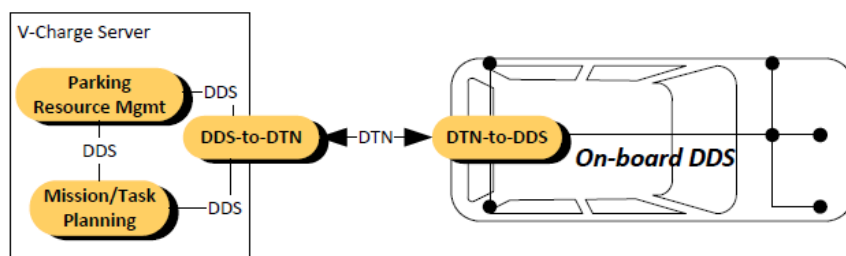


Figure 2 V2X architecture

Significant updates have been made to the Vehicle Interaction Concept (see Section 2.5 in D5.2), though. The interactions of the user with his vehicle and of the vehicle with the server are of utmost importance for final usability, safety, and user acceptance. The state machine in Figure 3 depicts the final version of this interaction process and the resulting communication protocol.

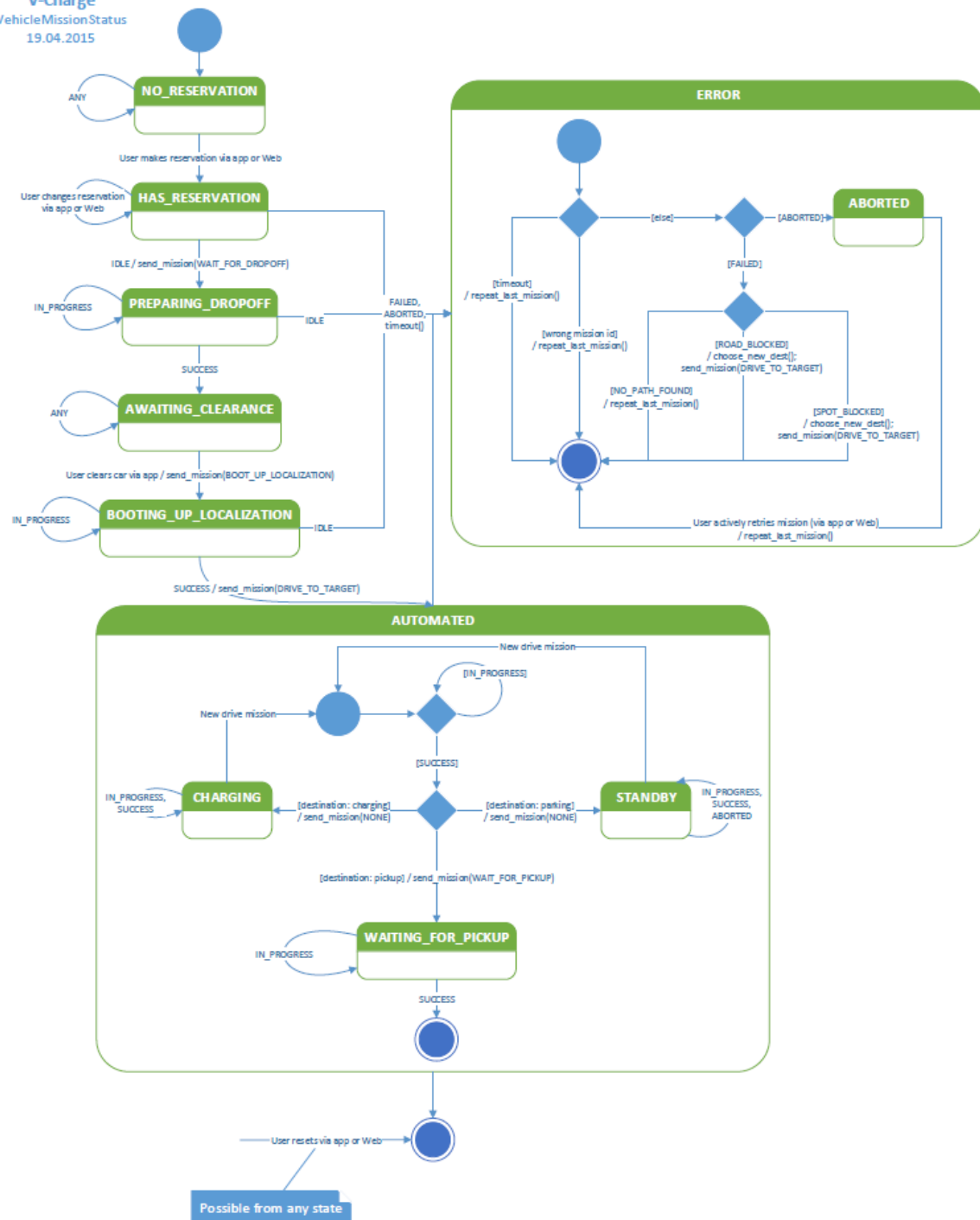


Figure 3 Full state machine for interaction between vehicles, the ParkingManager backend, and users.

The involved smartphone application has been thoroughly described in D5.4.

Results of the tests at MLC are shown in Figure 4. The blue bars depict the total number of trials for testing the drop-off functionality, the green ones show the number of successful trials. Since mid-April, the system is running stable and all tests have been successful.

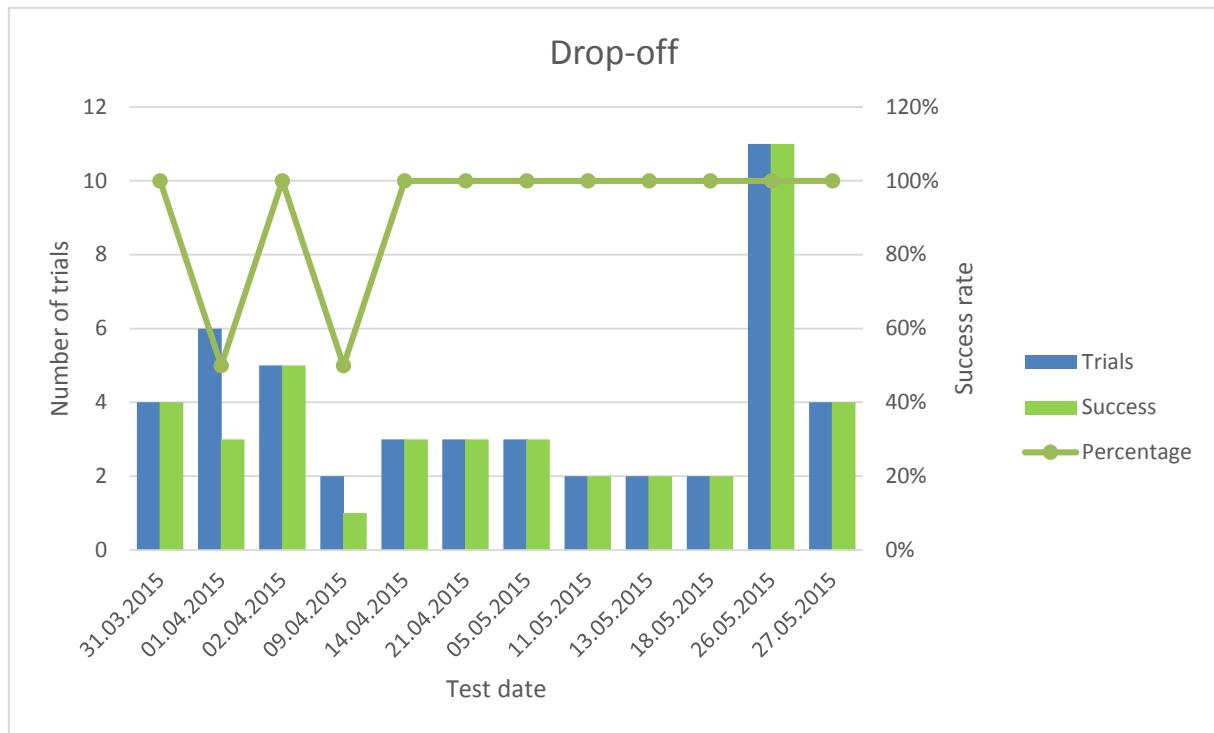


Figure 4 Results of testing Drop-Off

Scenario 2: Driver summons the car and deactivates the system

Maturity level: basic

Status: **ACHIEVED**

Success rate: 96% (23/24)

Main error source(s): Wifi/Mobile Link connectivity problems

Scenario specification

- robot is parked
- driver requests the vehicle through smartphone

Observable actions of the system

- robot drives through the parking area
- robot enters pick up zone and parks
- robot is deactivated

Necessary functionality

- HMI: driver to smartphone
- server to vehicle communication
- local path planning into pick-up zone
- system deactivation

Results

Using the same communication infrastructure as for the drop-off, the pick-up has also been thoroughly tested. As one can see, however, the total number of tests is lower than for the first interaction (drop-off) since the test of the pick-up functionality requires a completely successful test run up to this point. That means, if a test run had been aborted due to any failure (e.g., parking), the pick-up was not tested. Results are shown in Figure 5.

Due to the lower number of tests, this part of the system became stable a bit later than the drop-off (beginning of May).

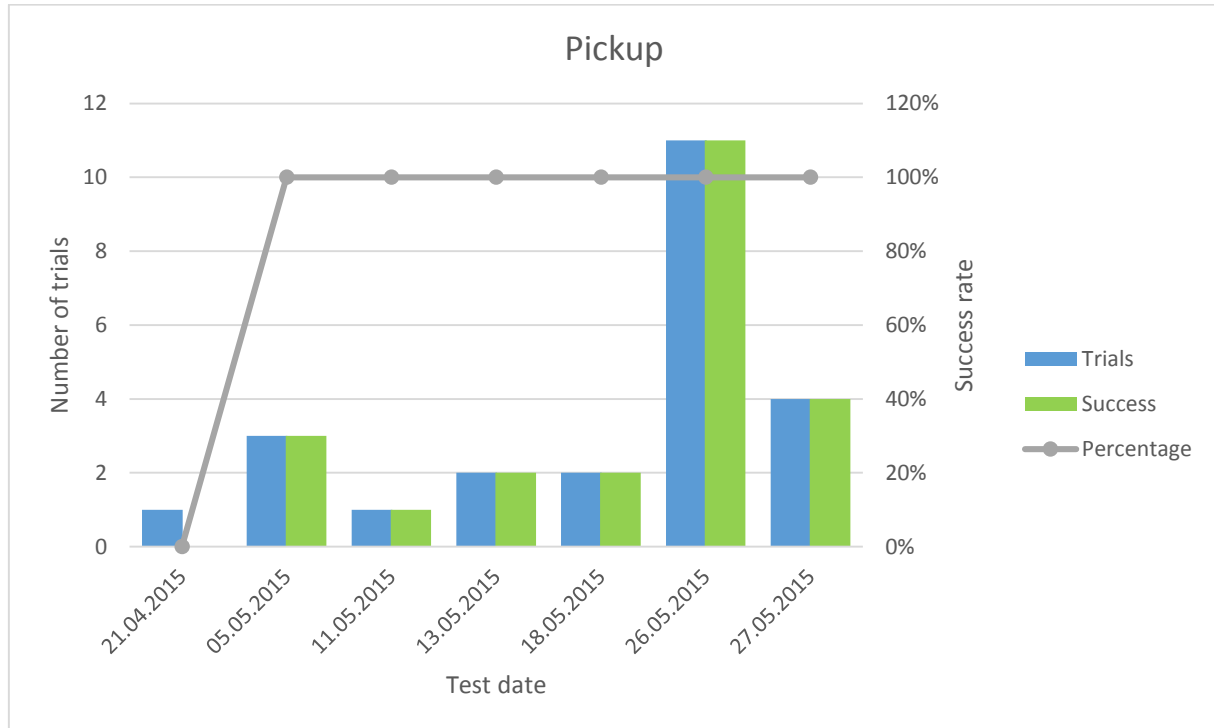


Figure 5 Results of testing pick-up functionality

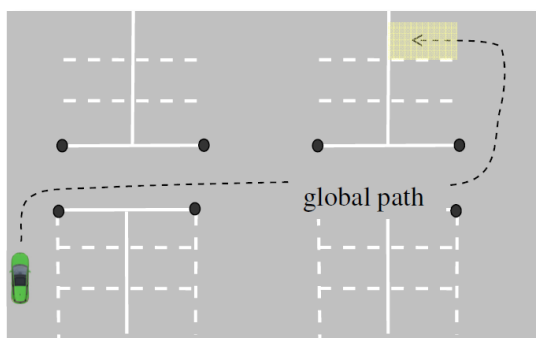
Scenario 3: Robot drives through parking area

Maturity level: basic

Status: **ACHIEVED**

Success rate: N/A

Main error source(s): None



Scenario specification

- no unexpected obstacles
- no traffic
- robot has knowledge of the infrastructure (parking spots, stop signs, landmarks)

Observable actions of the system

- robot drives through the parking area
- robot stops at stop signs
- robot uses indicators properly

Necessary functionality

- global path-planning
- path following
- localisation in empty parking area
- awareness of road topology
- global path

Results

This scenario necessitates basic navigation capabilities in a fully pre-mapped parking lot. Besides global path planning capabilities, the robot requires basic path following abilities. Respecting stop signs and using indicators properly requires awareness of the road topology and semantic information in it.

The main components involved in this scenario are:

- visual localization module
- semantic map
- global path planner
- local on-lane planner

Visual localization proved to be reliable and accurate and we refer to deliverable 3.5 for a detailed description and evaluation. The semantic mapping concept was laid out in deliverable 2.4.

Similar to Scenario 6: Robot passes an obstacle at the side of the lane, this scenario has not been tested individually. Obstacle-free navigation was operating on the vehicle for an extended period, rendering additional tests for this scenario obsolete. In the absence of any obstacles in the driving corridor around the reference path planned by the global planning module, the local planner acts as a model predictive controller minimizing speed offsets from a reference speed, lateral offsets from the reference path, steering effort and maximizing the distance to the closest obstacle. We refer to deliverable 4.3, section “2.2.1.1 Scenario 3: Obstacle-Free Navigation Along a Reference Path” for an evaluation.

As no stop signs were present at the test site at Mobile Life Campus, Wolfsburg, we refer to Scenario 13: Robot approaches and passes an intersection for a similar use-case where the vehicle has to stop at a certain location using semantic information of the environment.

The usage of indicators was implemented as part of the preparations for the final demonstration at Mobile Life Campus, Wolfsburg. Indicators were used to signal both the parking manoeuvres and turns at intersections. The module responsible for triggering the indicators uses 2 information items

from the global reference path and the semantic map: the road topology (forking of lanes, intersections, connections of lanes to parking spots) and the curvature of the lane segments.

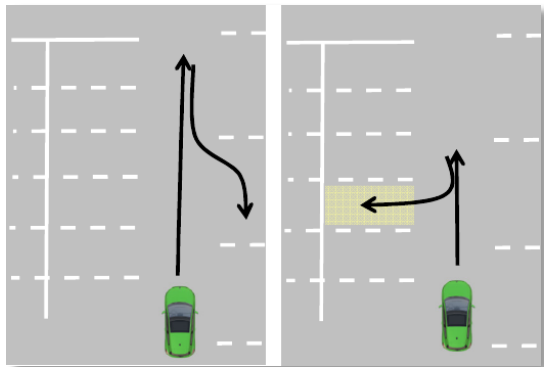
Scenario 4: Parking into and out of a parking spot or charging station

Maturity level: basic

Status: **ACHIEVED**

Success rate: 97% (259/267)

Main error source(s): Localisation errors, false positives in occupancy grid map



Scenario specification:

- no unexpected obstacles in path
- no traffic
- robot knows into which spot / charging station to park
- the spot / charging station is kept free by the system
- neighbouring spots might be occupied

Observable actions of the system:

- robot reverses parks into a free parking spot / charging station
- robot pulls out of parking spot / charging station

Necessary functionality:

- detection / measurement of the reserved spot / charging station
- local path planning into spot / charging station
- continuous localization relative to spot / charging station
- path following

Results

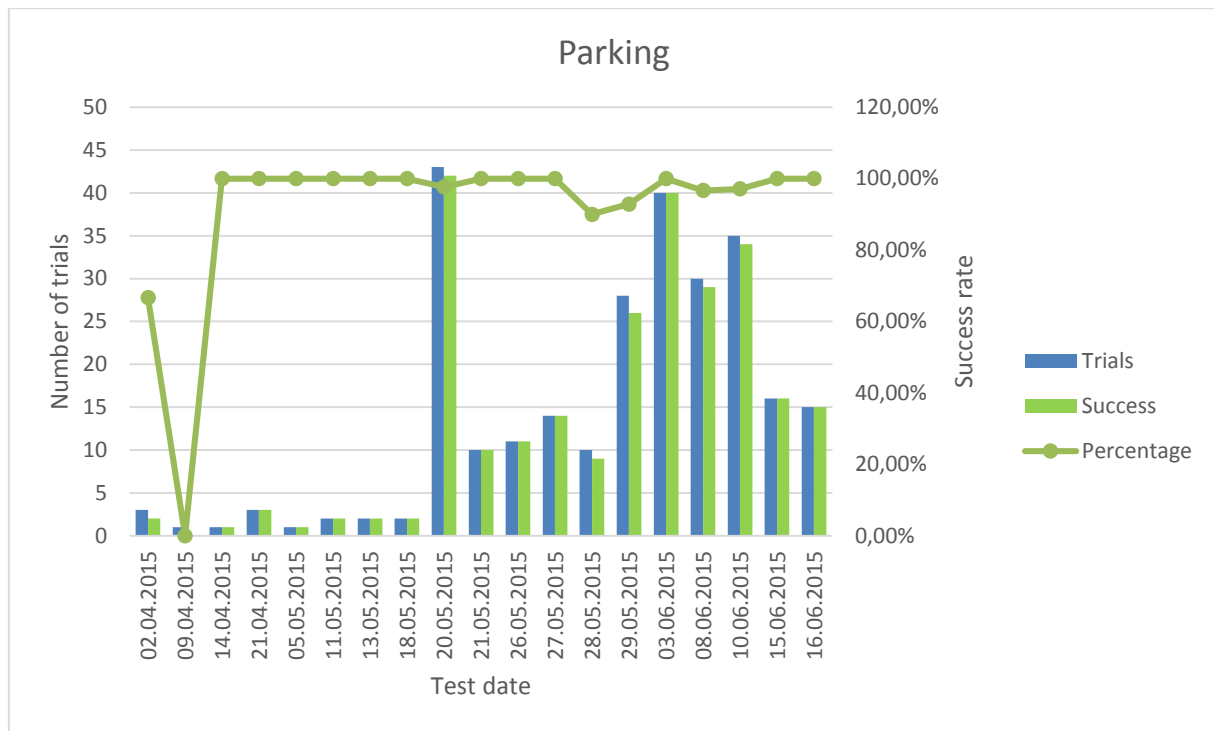


Figure 6 Statistics for parking scenario

In total 267 documented parking manoeuvres were performed. The evaluation includes backward parking as well as the docking to charging station scenario, which was performed as a forward parking manoeuvre. The results not only show great robustness (97% success rate) but also centimetre-level accuracy. Please refer to Deliverable 4.4 for the details of accuracy evaluation as well as analysis of main sources of error.

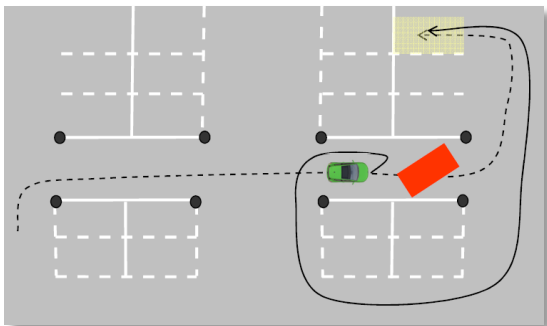
Scenario 5: Robot adjusts the route due to an obstacle blocking the road

Maturity level: basic

Status: **ACHIEVED**

Success rate: 94% (30/32)

Main error source(s): Deployment error of map management software



Scenario specification

- robot is on its way to parking / charging station

- an unexpected static obstacle blocks the way
- no traffic

Observable actions of the system

- robot stops
- robot continues the drive via a different route

Necessary functionality

- perception of static obstacles
- re-planning of global path
- local path-planning
- localisation in presence of static obstacles

Results

In the course of the project, this test scenario was slightly altered. Instead of demonstrating the vehicle's ability to re-plan in conjunction with an obstacle blocking its road, the global re-planning functionality was demonstrated in the case another vehicle blocks a pre-assigned parking spot.

The communication architecture developed for global re-planning was described in deliverable 4.3, section 2.1 "Global Task Planner (VW)".

This test scenario has shown very reliable performance as illustrated by Figure 7. The initial failed trials had their origins in the errors in the deployment procedure of maps.

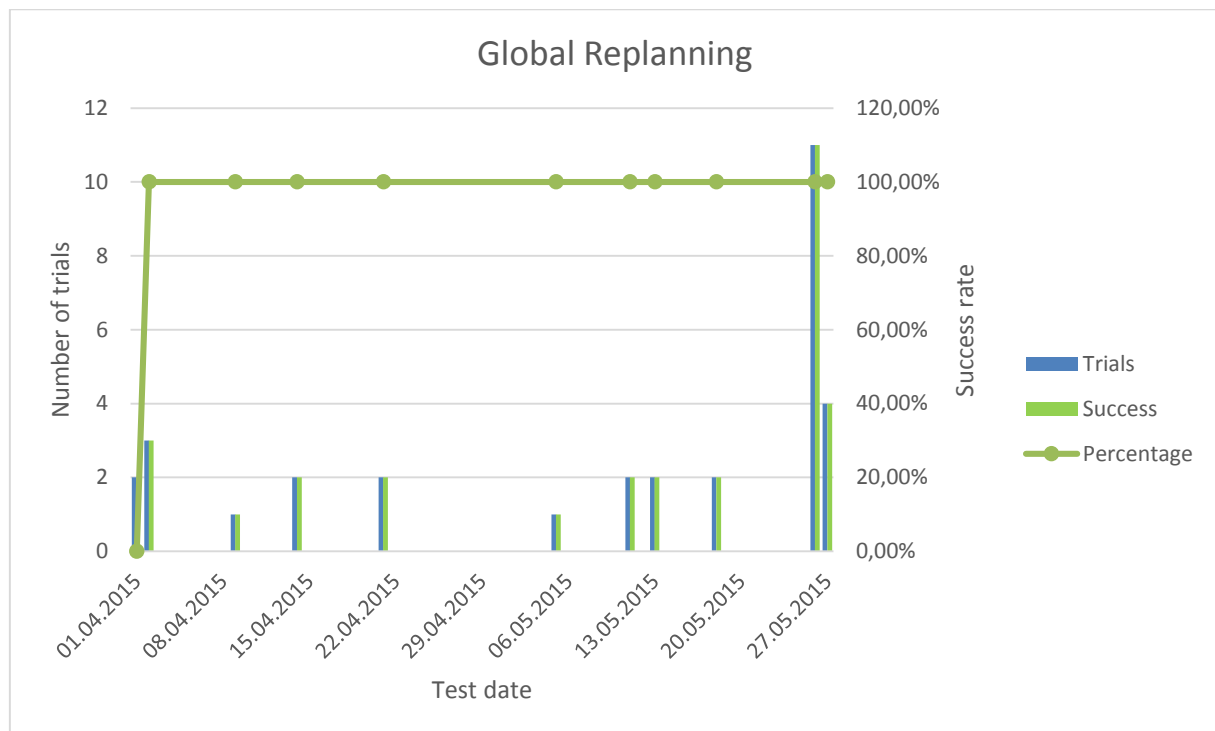


Figure 7 Statistics for global replanning scenario

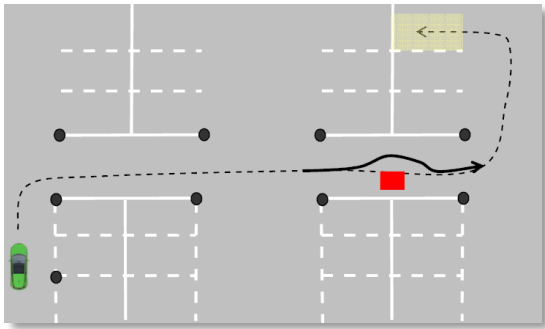
Scenario 6: Robot passes an obstacle at the side of the lane

Maturity level: basic

Status: **ACHIEVED**

Success rate: 100%

Main error source(s): N/A



Scenario specification

- robot is on its way to parking / charging station
- an unexpected static obstacle stands in the way but does not block it completely
- no traffic

Observable actions of the system

- robot slows down
- robot passes the obstacle
- robot returns to the middle of the lane

Necessary functionality

- perception of static obstacles
- scene analysis: can the obstacle be passed by?
- local path-planning for a passing manoeuvre
- localisation in presence of static obstacles

Results

The performance of the V-Charge system in this scenario depends on all modules listed under Scenario 3: Robot drives through parking area plus static environment perception modules. The static environment perception was evaluated in deliverable 3.5.

The statistics for this scenario are depicted in Figure 8. The scenario was only explicitly tested at the very beginning of the testing period, since the functionality was already operating for a long time on the vehicle and proved to be reliable since the very beginning. For an evaluation of the local planning framework in this scenario, we refer to deliverable 3.4, section 2.2.1.2 Scenario 6: Static Obstacle Handling.

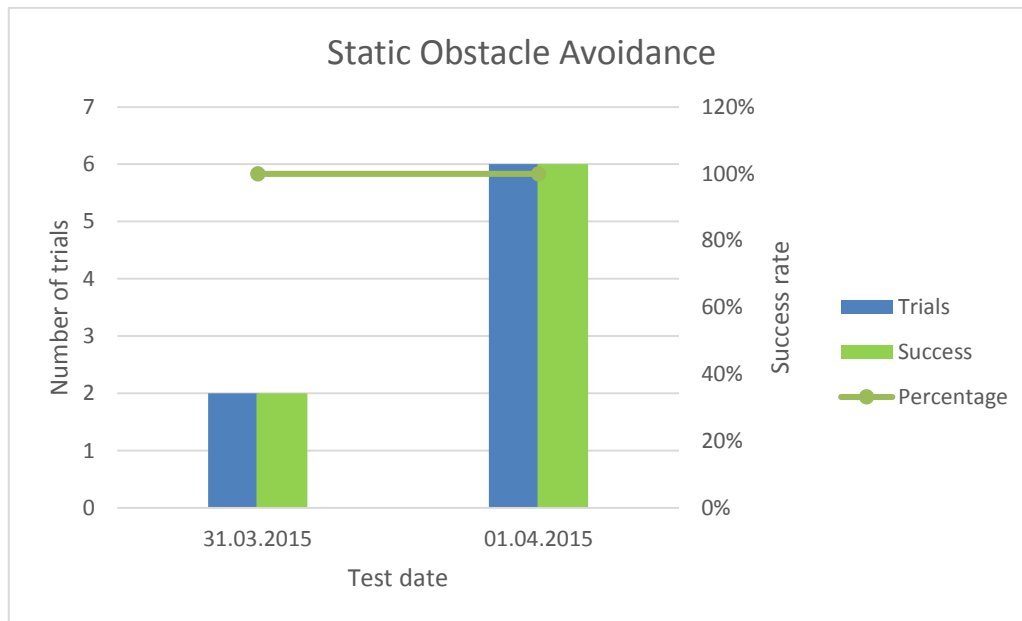


Figure 8 Statistics for static obstacle avoidance scenario

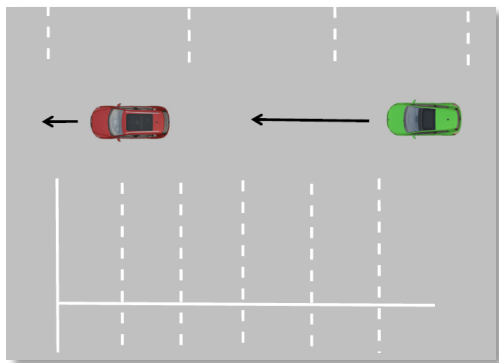
Scenario 7: Robot follows a vehicle in the front

Maturity level: reasonable

Status: **ACHIEVED**

Success rate: 79%

Main error source(s): Separation of static and dynamic information



Scenario specification:

- robot is on its way to parking / charging station
- a slow vehicle forces the robot to slow down
- no other obstacles in path
- no other traffic participants in vicinity

Observable actions of the system:

- robot slows down
- robot keeps safety distance

Necessary functionality:

- detection and tracking of dynamic objects

- situation interpretation: "following or passing?"
- adaptation of local trajectory
- localisation in presence of dynamic objects

Results

The approach and performance in this scenario is documented in detail in the publication "Vision-Only Fully Automated Driving in Dynamic Mixed-Traffic Scenarios" attached to deliverable 4.4. Figure 9 shows the test results again. The V-Charge vehicle reached a success rate of 79% in 33 test trials.

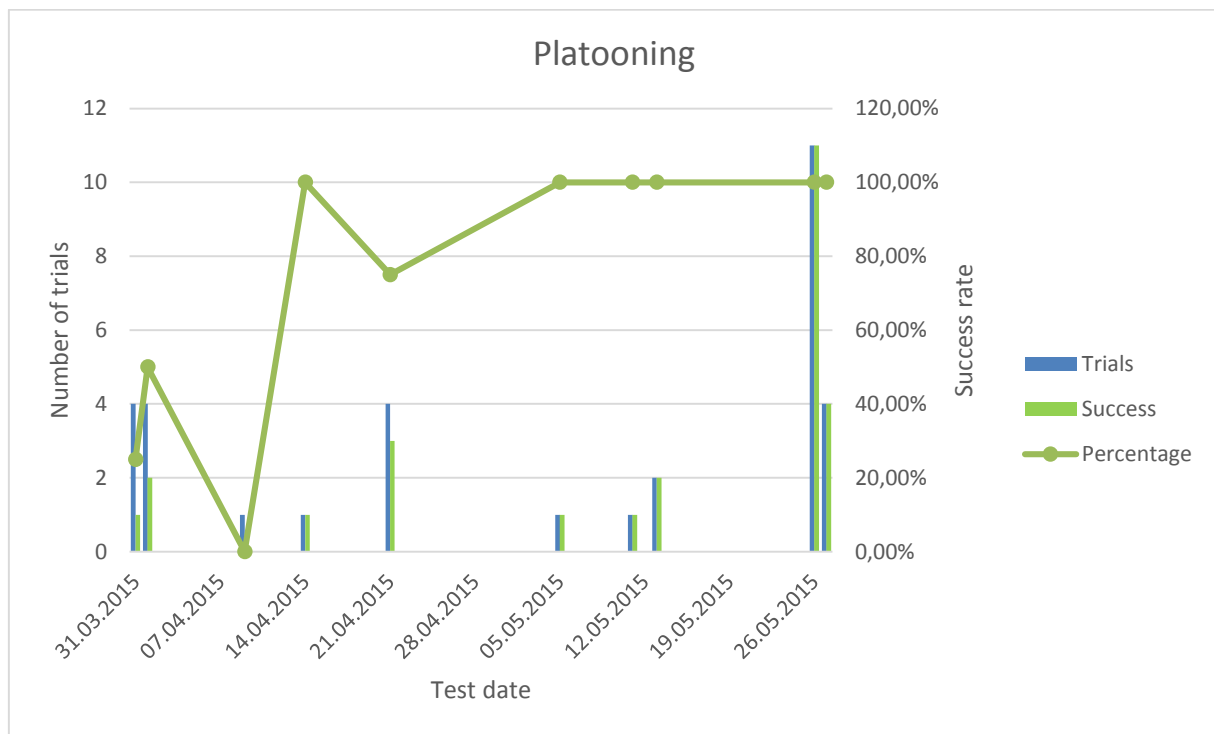


Figure 9 Statistics for platooning scenario

The main source of error in this scenario was the sporadically occurring problem of separating dynamic from static information, resulting in occupied pixels originating from the target vehicle in the gridmap. As a result, the V-Charge vehicles initiates avoidance manoeuvres. This behaviour is documented in deliverable 3.5 and 4.4 and depicted in Figure 10.

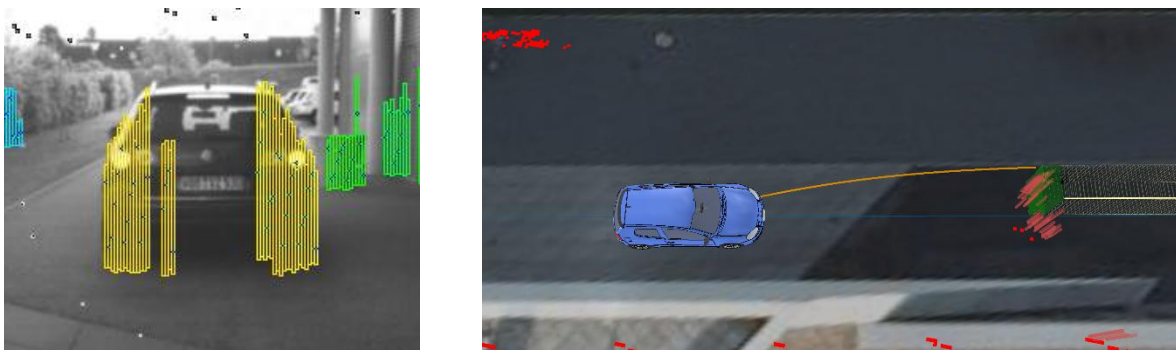


Figure 10 Sporadically occurring problem of separating static from dynamic information in the platooning scenario due to missing stixels from the front stereo camera

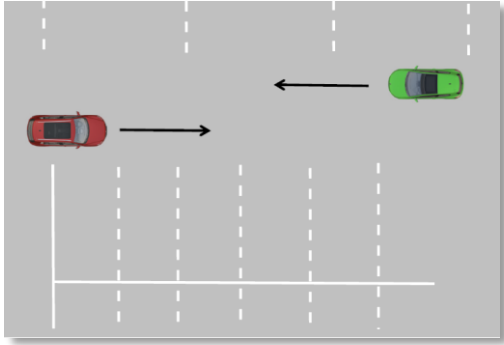
Scenario 8: Robot passes an oncoming vehicle

Maturity level: reasonable

Status: **PARTIALLY ACHIEVED**

Success rate: 70%

Main error source(s): late detection, noise in velocity estimates, over-conservative planning behaviour



Scenario specification

- robot is on its way to parking / charging station
- vehicle approaches the robot in its separate lane
- no other obstacles in path
- no other traffic participants in vicinity

Observable actions of the system:

- robot proceeds without reaction

Necessary functionality:

- detection and tracking of dynamic objects
- situation interpretation: "am I in danger?"
- localisation in presence of dynamic objects

Results

We consider this scenario as only partially achieved due to frequent braking manoeuvres of the ego vehicle. We account this shortcoming to noisier position and velocity estimates of the oncoming car, as well as sometimes inaccurate object predictions.

Distant objects are extremely small and hardly detectable in the image domain, due to the wide field of view of the fisheye camera. Moreover the amount of available frames while the vehicle can still react and avoid the collision are few because of the high relative speed.

In other scenarios – like following another vehicle or avoid crossing pedestrians – we exploit the tracking to increase the resulting reliability and accuracy. Since we have significantly fewer frames available in this scenario we can't do the same. This is why position and velocity estimates are noisier in this scenario than in others.

Inaccurate object predictions stem mostly from a practical problem. At the testing site at Mobile Life Campus, Wolfsburg, only one location contained lanes with allowed opposite directions. In this area

however, the notion of lanes was difficult to apply, since it was comprised of open space area, making accurate object predictions utilizing lane information difficult.

The performance of each perception system used in this scenario is documented in Deliverable 3.5. Figure 11 visualizes the statistics obtained during the scenario tests. The scenario was tested several times with a successful rate of about 70%, but it is important to point out that we consider a test only successful if the robot acts flawlessly from a journalist's perspective: while the car stopping unnecessarily as a result from conservative behaviour due to a noisy detection is comprehensible in general, it may however not be a transparent reaction for a non-technical person. The decreasing number of tests for this scenario can be accounted to the fact that this scenario was removed from the final demonstrations as it was not satisfactory handled and less media-effective.

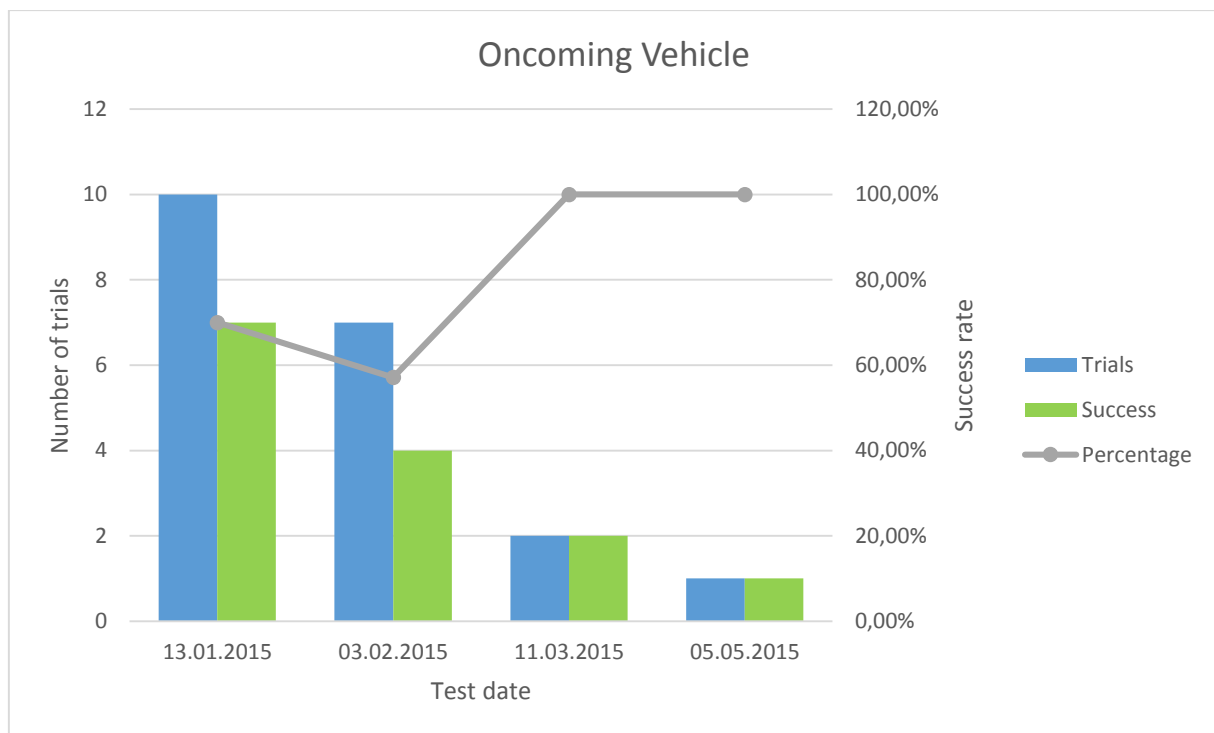


Figure 11 Statistics for oncoming vehicle scenario

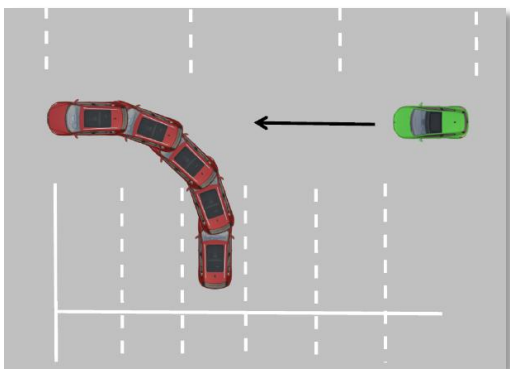
Scenario 9: Robot lets other vehicle pull out of a parking spot

Maturity level: reasonable

Status: **NOT TACKLED**

Success rate: N/A

Main error source(s): accurate velocity estimation of partially occluded objects



Scenario specification

- robot is on its way to parking / charging station
- way is blocked by a vehicle pulling out of a parking spot
- no other obstacles in path
- no other traffic participants in vicinity

Observable actions of the system

- robot slows down / stops
- robot provides enough space for the manoeuvre of the vehicle
- robot continues the drive

Necessary functionality

- detection and tracking of dynamic objects
- situation interpretation: "shall I wait or drive a different path?"
- local trajectory adaptation
- localisation in presence of dynamic objects

Results

Due to a limited time budget, the V-Charge team decided not to tackle this scenario. While the combination of the local motion planning and object prediction framework is readily in place to handle this situation (see deliverable 4.3 for a test of this scenario with the off-the-shelf V-Charge software), the difficulty of reliably estimating other object's velocities prevented this scenario from further being pursued. The low-cost camera-based sensor setup of the V-Charge vehicle renders it difficult to distinguish between statically parked cars and vehicles in the progress of pulling out of the parking spot. While accurate velocity estimation is already challenging for non-occluded objects, occlusion quite common for cars parked in rows renders it even more difficult. Fully enabling the local motion planning suite in this scenario would yield frequent braking manoeuvres when navigating through narrow parking rows. Appealing navigation performance in parking rows was achieved by utilizing a strong zero-velocity prior for vehicles detected on top of parking spots, sacrificing the capability of the V-Charge car to perform this scenario at hand.

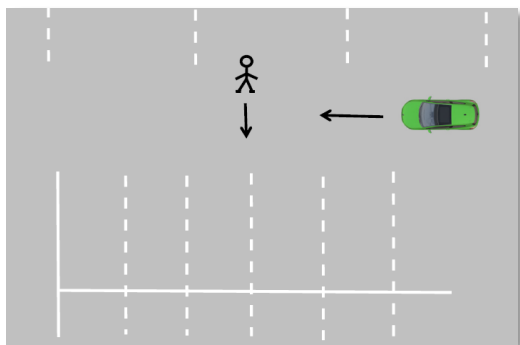
Scenario 10: Pedestrian is crossing the road in front of the car

Maturity level: reasonable

Status: **ACHIEVED**

Success rate: 74%

Main error source(s):



Scenario specification

- robot is on its way to parking / charging station
- pedestrian crosses the robot's path
- no other obstacles in path
- no other traffic participants in vicinity

Observable actions of the system

- robot slows down / stops
- robot lets the pedestrian cross the lane
- robot continues the drive

Necessary functionality

- detection and tracking of dynamic objects
- situation interpretation
- local trajectory adaptation
- localisation in presence of dynamic objects

Results

The performance of the system in this scenario is documented in deliverable 4.4, including success statistics collected over the test trials. These results are restated in Figure 12. Overall the performance of the V-Charge car was excellent in this scenario. The diminished performance of the system during one test day was caused by a software bug in the classification module, resulting in a large false negative rate. These trials are not representative for the overall performance in this scenario, yet are included for the sake of completeness of this report.

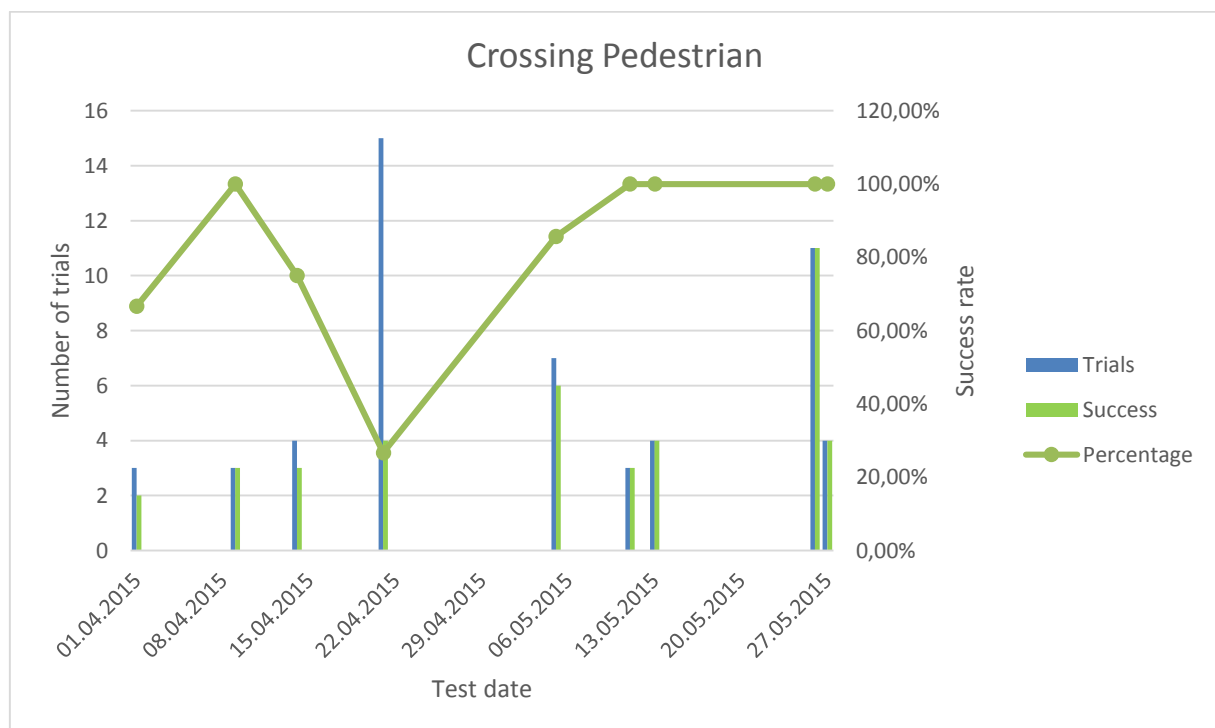


Figure 12 Statistics for crossing pedestrian scenario

The good performance of the vehicle during this scenario can – besides the flawless operation of the modules involved – be accounted to the good velocity estimation for pedestrians. For pedestrians,

the stixel count is rather low and shape estimation becomes an easier task. Stable geometry estimation leads to less fluctuation in the velocity estimates.

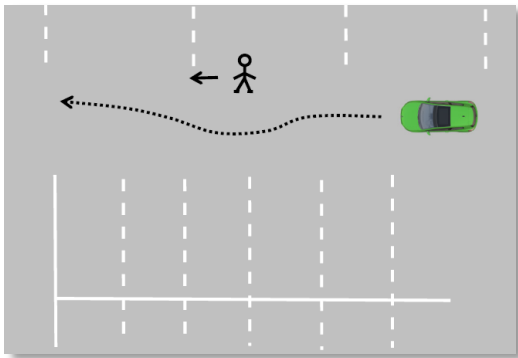
Scenario 11: Robot passes a pedestrian walking at the side of the road

Maturity level: exceptional / outstanding

Status: **PARTIALLY ACHIEVED**

Success rate: N/A

Main error source(s): noise in velocity estimation, over-conservative planning behaviour



Scenario specification

- robot is on its way to parking / charging station
- pedestrian walks along robot's lane
- no other obstacles in path
- no other traffic participants in vicinity

Observable actions of the system

- robot slows down
- robot passes the pedestrian whilst keeping safety distance at all times
- robot returns to the middle of the lane

Necessary functionality

- detection and tracking of dynamic objects in the monocular cameras
- scene analysis: is there enough space to pass by?
- local path-planning for a passing/overtaking manoeuvre
- localisation in presence of dynamic objects

Results

The main challenges of this scenario are on the one hand the robust detection and tracking of the pedestrian in the monocular cameras, also in the regions not covered by the front-looking stereo sensor. On the other hand, motion planning in close proximity to other agents requires precise forecasting of their future actions, which is challenging to do given only limited information (no high-level contextual information available, e.g. where the pedestrian is looking at) and noisy position and velocity estimates.

The first challenge of object detection and tracking from monocular cameras was tackled by Parma. The correct detection rate of the pedestrian classifier during this scenario is 95.9%. In order to estimate the pedestrian position we perform the tracking jointly through the four monocular cameras.

Pedestrians in the vehicle corners can be seen from two cameras and in front/rear of the vehicle are in the field of views of the stereo systems, but while the vehicle is overtaking the pedestrian only the monocular lateral cameras are able to detect them for some time.

The following table shows the standard deviation of the pedestrian position estimation related to the number of point of views from which is visible:

	Detected by two cameras	Detected by one camera
Standard deviation of position	12.25 [cm]	39.59 [cm]

Table 1 Standard deviation of position estimates of pedestrians from monocular cameras. Note: the pedestrian was standing in the direct surrounding of the vehicle.

As expected, when the sources of information are only the side cameras, the standard deviation of the pedestrian position estimate is higher because we can rely only on the ground plane assumption. Consequently velocity estimates get noisier.

As detailed in Deliverable 4.4, the local motion planner reacts overly conservative, which is especially notable during navigation in close proximity to pedestrians. The purely reactive navigation scheme leads to frequent braking manoeuvres due to noisy velocity estimates.

Due to these problems, the V-Charge team decided not to show this scenario during the final demonstrations, and consequently removed it from the tight testing schedule. Yet the pipeline is fully implemented and operational, why we consider this scenario as partially achieved. Visual results can be seen in the video attachment `V-Charge_amsterdam_monodetection.mp4`.

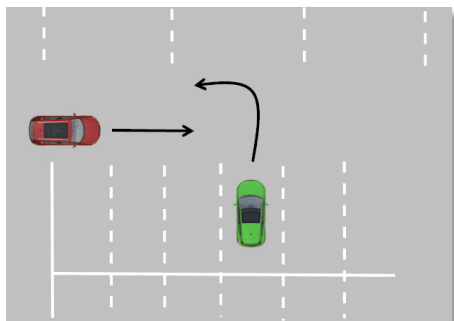
Scenario 12: Robot pulls out of parking spot in presence of traffic

Maturity level: exceptional / outstanding

Status: **NOT TACKLED**

Success rate: N/A

Main error source(s): N/A



Scenario specification

- robot is about to pull out of parking spot and to join the traffic
- a vehicle approaches from the left or right
- maximum one vehicle in vicinity at any time

- no other obstacles in the way

Observable actions of the system

- robot signals the intention of joining traffic
- robot waits until manoeuvre can be conducted safely
- robot pulls out and drives in the correct lane

Necessary functionality

- dynamic object detection to the left and right of the robot
- situation analysis: "how much time do I have, how much time do I need?"
- path planning and following
- localisation in presence of dynamic objects

Results

From a perception point of view this scenario proved to be very challenging. In order to be able to see the oncoming car, our own car needs to be standing at the very front right next to the street. Depending on the parking manoeuvre that was performed earlier we first need to drive to the border of the parking spot to see the oncoming car at all and even then it is only visible at the border of the image in one single camera. On the border of the image the resolution is very low. This makes it very difficult to detect the car and predict its velocity. Also, there is no sensor available on our platform that can directly measure the distance to the oncoming car.

We developed functionality in workpackage 3 and 4 to enable the robot to safely pass intersections. In case our car is standing still, all movements in the scene must stem from dynamic objects. For this scenario we specifically developed and integrated an algorithm exploiting its characteristic: while the vehicle is standing all the moving features in the scene are related only to dynamic obstacles. To detect these obstacles the features are clustered, then each cluster is grouped with its neighbours and tracked over the time; subsequently the three different point of views (left, front and right cameras) are exploited to increase both the accuracy and reliability.

In general, a similar approach is also suitable for handling the scenario in this section. We believe that the approach, coupled with semantic information from the semantic map, would provide the robot a sufficient amount of information to reason about the pull-out action. To adopt the insights gained from scenario Scenario 13: Robot approaches and passes an intersection, one would have to script the behaviour of the vehicle during pull-out: firstly, from the accurate gridmap environment representation built up during pull-in one can compute an adequate position of the car to observe the orthogonal lane. An intermediate target position for the parking planner can be introduced in order to move the vehicle to the location where it is able to monitor the incoming lane and proceed only in case of clearance.

However the approach described above is purely conceptual and was not implemented on the platform for this scenario due to time constraints.

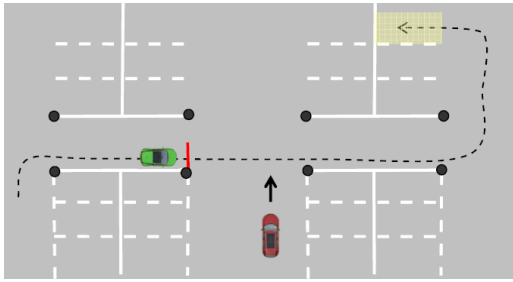
Scenario 13: Robot approaches and passes an intersection

Maturity level: exceptional / outstanding

Status: **ACHIEVED**

Success rate: 92% (22/24)

Main error source(s): N/A



Scenario specification

- robot approaches an intersection
- another vehicle approaches from the front, left or right
- maximum one vehicle in vicinity at any time
- no other obstacles in the way

Observable actions of the system:

- robot stops at stop-line
- robot yields the right of way (if necessary)
- robot continues driving

Necessary functionality:

- knowledge of traffic rules and road topology
- dynamic object detection in front of and to the left and right of the robot
- situation interpretation: "how much time do I have, how much time do I need?"
- path planning and following
- localisation in presence of dynamic objects

Results

The functionality for performing this scenario was fully implemented and deployed to the vehicle. The functionality was tested during the run-up to the final demonstration at Mobile Life Campus, Wolfsburg and demonstrated at this.

The modules involved in this scenario are:

- Dynamic object detection at intersections (Parma, WP3)
- Intersection task processor (ETHZ, WP4)

The functioning and results for the dynamic object detection at intersections was presented in deliverable 3.5, yielding a true positive detection rate of 95.9% for pedestrians and 98.1% for vehicles. Provided these object detection and classification results, ETHZ implemented a separate module (task processor) to monitor the clearance of an intersection. In order to deal with the significant noise levels in both position but especially velocity estimation of the dynamic objects from the monocular cameras, a separate module from the default on-lane local planning module was chosen to better adapt to this detection characteristics. The intersection task processor uses information from the semantic map in order to match objects to the lanes and to filter irrelevant objects on outgoing lanes using directed connectivity information from the semantic map. The module first stops the car at an appropriate location at the intersection, enabling the dynamic object detection at intersections to measure nearby objects. After 5 seconds of no observed activity at the intersection, the module hands over to the standard local on-lane planner.

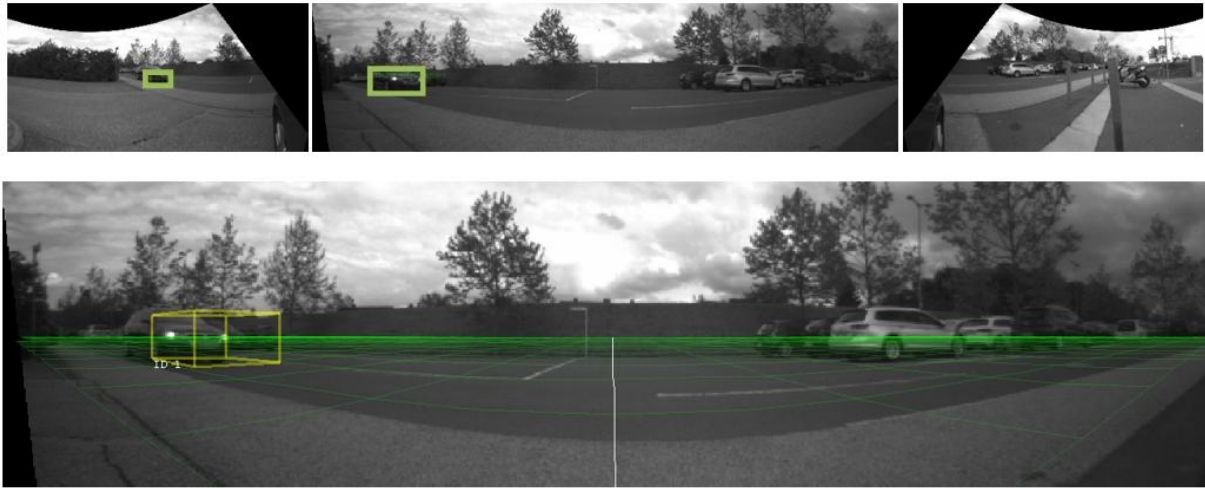


Figure 13 Dynamic Object Detection at Intersections

Figure 14 shows the statistics collected for this scenario. The V-Charge vehicle completed 22 out of 24 trials successfully.

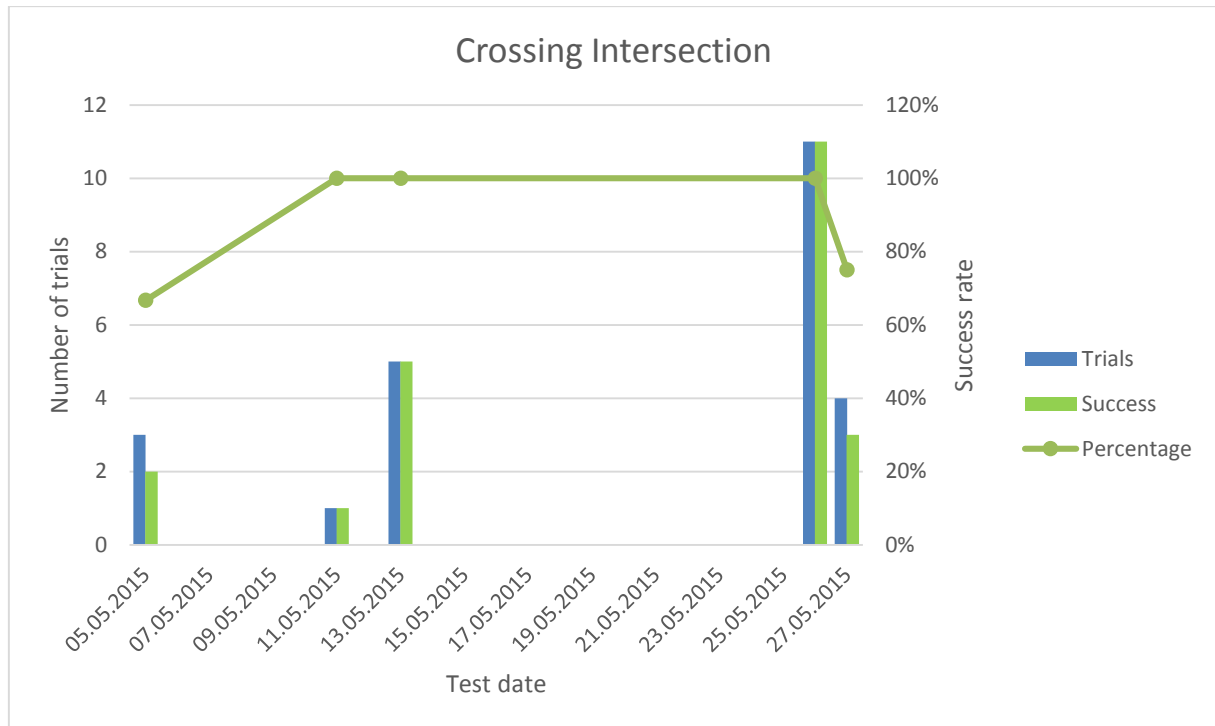
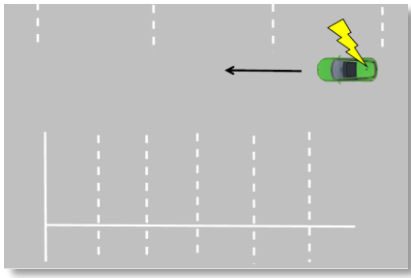


Figure 14 Scenario statistics for crossing intersection

Scenario 14: Robot reacts to an emergency signal

Maturity level: basic

Status: **ACHIEVED**



Scenario specification:

- robot is in autonomous mode – in any phase
- emergency signal is given

Observable actions of the system:

- robot performs an emergency brake
- robot deactivates itself
- problem is solved
- robot is manually reactivated

Necessary functionality:

- safety system is integrated into vehicle
- emergency brake
- system and mission reactivation

Results

The safety system in V-Charge has been developed in cooperation with TÜV Süd, is multi layered and is presented in more detail in Deliverable D1.3 Section 3 as well as Deliverable D8.4, Task 1.6. Here we recap its most important elements:

1. Safety driver in passenger's seat in case of test runs. Safety driver's actions (braking, steering) lead to immediate disconnecting of computer system from the vehicle system.
2. Remote kill switches in case of presentation runs. Pressing the "KILL" button leads to immediate applying of the brakes (including the parking brake) and disconnecting of computer system from the vehicle system. The same reaction takes place in case of lost connection between kill switch transmitter and receiver or interruption of power supply of the kill switch receiver.
3. Additionally: multi layered software watchdog applying reaction as described in point 2 in case of the following events:
 - errors in vehicle ECUs
 - exceeding maximum allowed speed
 - opening vehicle doors
 - malfunctioning of a software or hardware component
 - distance to obstacles below a certain threshold as observed by so called virtual safety bumper

Throughout the project lifetime the safety concept has proven to be 100% effective. In case of points 1 and 2 there was not a single event of malfunctioning (here we ignore the false positives – i.e. undesired "killing" of the vehicle in case of broken connection between the kill switch receiver and

transmitter). From within the software-based watchdog the virtual safety bumper deserves special praise. It has proven very useful and effective: many journalists would “jump in” towards the car; the analysis of some data logs has shown that in many instances the reaction of the safety bumper was quicker than that of the safety person.

Summary and Conclusion

In this document, the performance of the V-Charge vehicle was documented based on 14 predefined test cases.

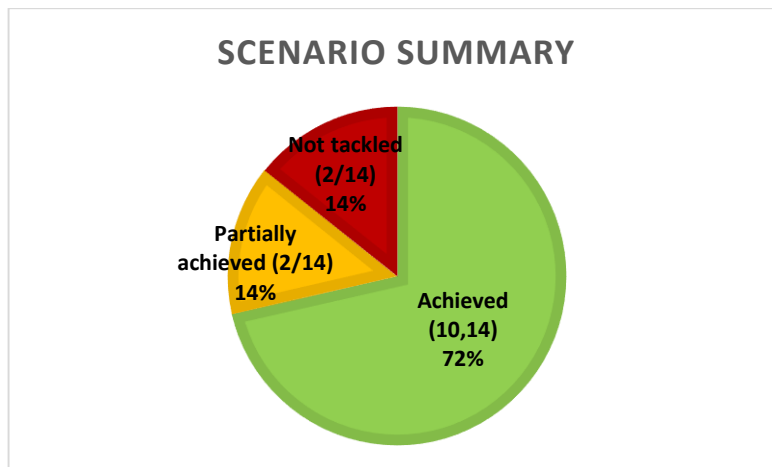


Figure 15 Scenario summary

As depicted in Figure 15, ten of them were fully achieved. Two of them (Scenario 8: Robot passes an oncoming vehicle and Scenario 11: Robot passes a pedestrian walking at the side of the road) were fully implemented and tested, yet not to our satisfaction. Two scenarios (Scenario 9: Robot lets other vehicle pull out of a parking spot and Scenario 12: Robot pulls out of parking spot in presence of traffic) were neither tackled nor tested.

The V-Charge system proved itself to be fully capable of navigation in static environments. Drop-off, pick-up, on-lane planning and parking were reliable given environment perception and localization in these scenarios.

Navigation in dynamic scenarios is improvable in some scenarios. Two main shortcomings turned out to be accurate velocity estimation and interaction-aware motion planning. The purely reactive motion planning framework was vulnerable to noise in velocity estimates of other agents navigating in close proximity to the ego vehicle. Here, more precise velocity estimates and a motion planning approach capable of considering other agents' reaction to the ego vehicle will turn out beneficial.