TrimBot2020 Deliverable D7.5

Demonstrator 1: Vehicle Localisation and Navigation

Principal Author: UEDIN
Contributors: UEDIN, BOSCH, ETHZ
Dissemination: CO

Abstract: This demonstrator tested the ability of the vehicle to localize itself based on the user sketch map, 3D data and previous 3D SLAM-based representations of the garden. The robot navigated from random initial positions to sketch-map specified locations and aligned itself with the nearby hedges or bushes via local visual servoing.

We have evaluated the performance of the robot during navigation in the garden, primarily in terms of routing and goal accuracy, including break-down to individual components, and compared the results to the demonstrator and evaluation plans.

The results show that all components can work well together to accomplish the primary goal of navigation without collisions. In some situations the navigation was however unstable and not accurate enough to meet given limits. We have identified multiple hard challenges in the Renningen test garden, which cause difficulties for most visual SLAM algorithms, and will be addressed in the next updates of the visual components.

Deliverable due: Month 24

1 Introduction

This report will first present how the demo event proceeded, what data was used for the evaluation, analysis of derived characteristics and comparison to previously given requirements. Finally, the demo success will be discussed based on criteria from *D7.2 - Demonstrator Plan*, ie. whether:

- Vehicle sensor system can acquire accurate 3D data of the garden,
- Vehicle can navigate on grass, loose soil, pavement,
- Vehicle can estimate its location with a target accuracy of 5-10 cm, and
- Vehicle can navigate to specified locations in the garden near hedges, and bushes without collisions.

In particular, the following capabilities were tested:

- 3D scene data capture
- User map-based target specification
- 3D-to-map registration
- Vehicle map-based self-localisation
- Point-to-point map-based path planning and dead-reckoning navigation
- Obstacle detection and avoidance

The original plan also included 3D surface recovery and fusion, but these were not found to be critical for the success of the first demo and the integration of corresponding components was postponed. Instead of the dense result expected by the demonstrator plan, we have decided to use the sparse 3D map from SLAM and evaluate its accuracy.

2 Procedure

The functionality of the Demonstrator 1 system comprised of the following tasks:

1. User sketches garden (display of map).
2. User indicates a point near hedge or topiary bush.
3. Vehicle navigates, while avoiding obstacles, to specified locations across a variety of terrain (observation of motion).
4. Vehicle servos (using the motion module) based on a 3D scene descriptions (SLAM map) to specified target location (based on the user sketch map).

These task were demonstrated at the review meeting of the project in September 2017 at the Bosch test garden in Renningen, Germany. The first task (garden sketch) was prepared in advance and presented on a screen. The remaining tasks were performed during a live event (Fig. 1).

Figure 1: The TrimBot2020 prototype (Platform 2) navigating the garden during live demonstration.

3 Integrated Components

The following components listed in *D7.1 - System Requirements Document* were integrated:

- Vehicle base (BOSCH)
- 3D sensor mounted on vehicle (ETHZ)
- Navigation training garden (BOSCH)
- 3D-to-map deformable registration algorithm (UEDIN)
- Semantic SLAM and relocalisation (ETHZ)
- User sketch map and interface (UEDIN)
- Master controller (BOSCH)
- Navigation execution (BOSCH)
- Vehicle control (BOSCH)

Some of the integrated components are still under development and certain features were not yet fully implemented: the map registration is currently only rigid and the SLAM component is currently not using semantic information. Details about the integration process can be found in *D6.2 - Integrated demonstrator 1.*

Figure 2: Bosch test garden (left) and its ground truth point cloud.

4 Evaluation

This section describes concrete data and methodology used to evaluate the principal characteristics of the evaluated components.

4.1 Ground Truth Data

We have used the GT data for the BOSCH garden as described in *D7.4 - Ground-truth data definitions and acquisition*, acquired before the demo event. The garden geometry was captured with a stationary Leica laser scanner as a point cloud file (Fig. 2). The scanner accuracy is around 3 mm and the point cloud merged from multiple locations was uniformly resampled at 10 mm. Several markers with reflectors were fixed in the garden to establish coordinate reference for subsequent tracking with Topcon laser tracker.

4.2 Calibration

Camera rig calibration of the sensor was performed using the Kalibr toolbox[1], details can be found in *D3.1 - Data representation design and implementation, sensor calibration.* Based on the report produced by Kalibr the mean re-projection error of calibration targets was below 1 pixel.

4.3 Mapping

We have evaluated the quality of the 3D reconstruction (Fig. 3b) based on accuracy of the reconstruction, i.e., how accurately the 3D mesh models the scene. Following the usual evaluation methodology described in [2], accuracy is distance d (in m) such that 90% of the reconstruction is within d of the ground truth.

The distances between the reconstruction and GT shown in Fig. 3a are calculated using a cloud-to-cloud metric, with local 2.5D surface estimation[2]. Cold colors indicate well reconstructed segments while hot colors indicate noisy parts and outliers. Figure 3 shows the results for the complete garden.

[1] https://github.com/ethz-asl/kalibr/wiki

[2] http://www.cloudcompare.org/doc/wiki/index.php?title=Distances_Computation

The evaluation was limited to the space delimited in XY by the perimeter of the garden area (surrounding walls) and in Z to a 1 m high section above the ground, ie. lamp-tops were excluded, shown in Fig. 4. Following [1] we also plot cumulative histograms of distances in Fig. 4c. The primary source of error was most likely in the weak texture and the reflective surfaces on the surrounding walls (windows, doors, metal plates).

4.4 Obstacle Detection

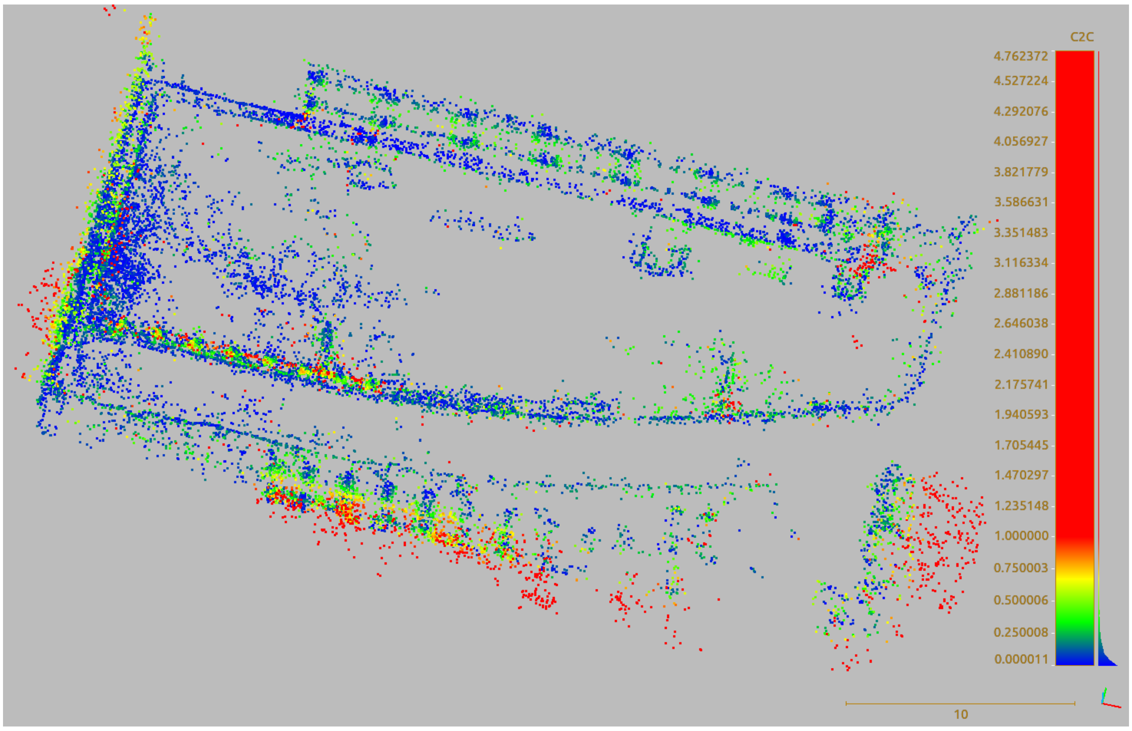
In the initial phase of the project the scene understanding components (WP4) are not yet integrated to detect obstacles. As a workaround we have decided to rely on 3D geometry of the acquired SLAM map (Sec.4.3), which was pre-processed to remove isolated points using statistical outlier filter[3]. The first step was to use the 3D SLAM height map (Fig. 5a) to derive obstacle probability (Fig. 5b). The probability measure was calculated from height after the the ground plane height estimate was subtracted. In the second step a suitable threshold was empirically chosen to produce a binary obstacle grid (Fig. 5c). The sparsity of the map allowed us to calculate the grid with resolution of 10 cm, which should improve when we use dense reconstruction instead.

4.5 Registration

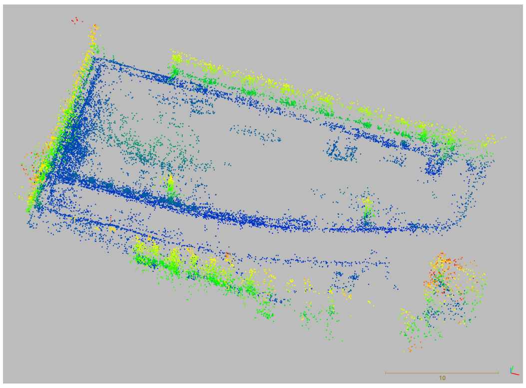
The GT registration between the SLAM point cloud and the GT point cloud was estimated from several correspondences of points which can be identified in both, eg. building corners and centers of lamps and bushes. The correspondences were manually assigned in CloudCompare[4], which computed a similarity transform (rotation, translation and scale). The accuracy of the manual registration was estimated from multiple (5) trials, with standard errors of the mean (SEM) translation 6.3 cm (max 12 cm) and rotation 0.1 deg (max 0.45 deg). The scale of the SLAM map (based on camera calibration parameters) was different from the GT by the factor of 0.9, which can be attributed to drift accumulation in the SLAM map resulting in non-metric deformations at large scale.

The sketch map was registered from manually input correspondences between a subset of sketched objects and locations in the 2D map of detected obstacles (Fig. 5b). Correspondences were used to estimate a 7 DOF transform (rotation, translation and scale), which was applied to the sketch map and combined with the detected obstacles (5c) to produce the occupancy map (Fig. 5d) for use with path planning and obstacle avoidance during navigation. The accuracy of this registration was estimated from multiple trials (5) and compared with above estimated GT transform, resulting in error of translation 6 cm and rotation 1.8 deg.

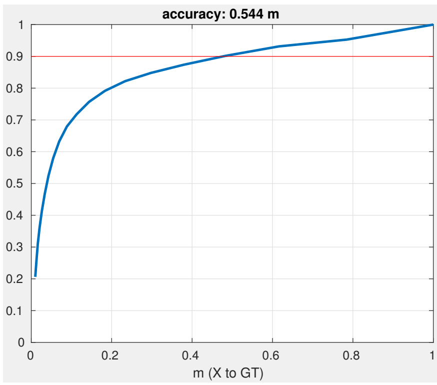
[3] http://www.cloudcompare.org/doc/wiki/index.php?title=SOR_filter
[4] http://www.danielgm.net/cc/



(a) Distance (m) of reconstructed point cloud to the GT reference

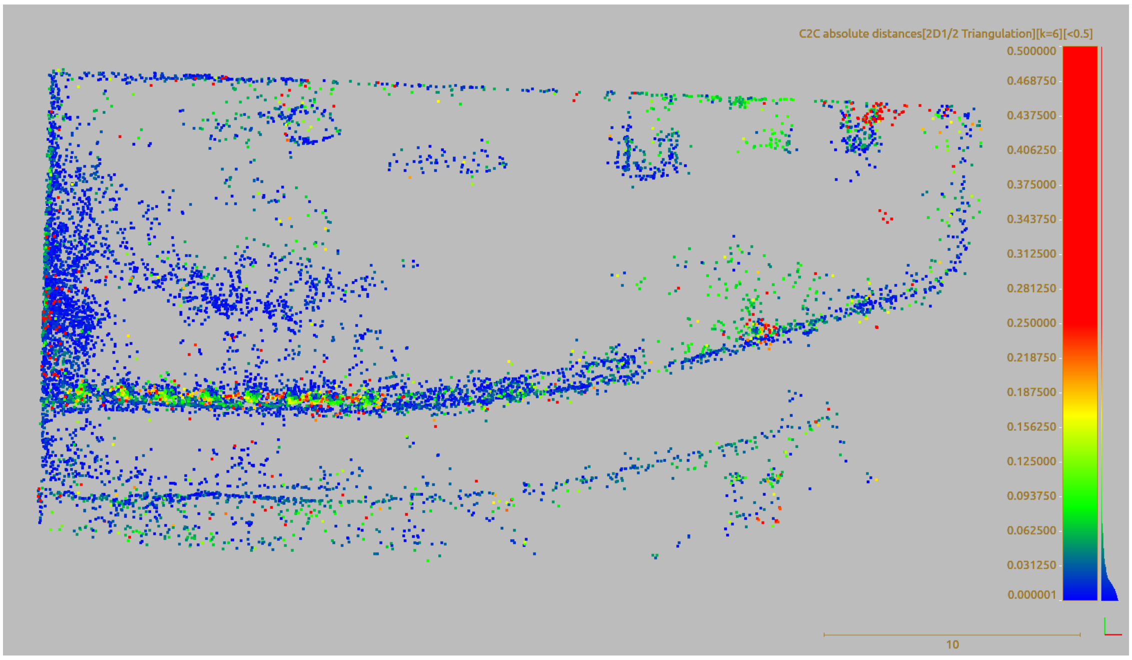


(b) Height-colored point cloud

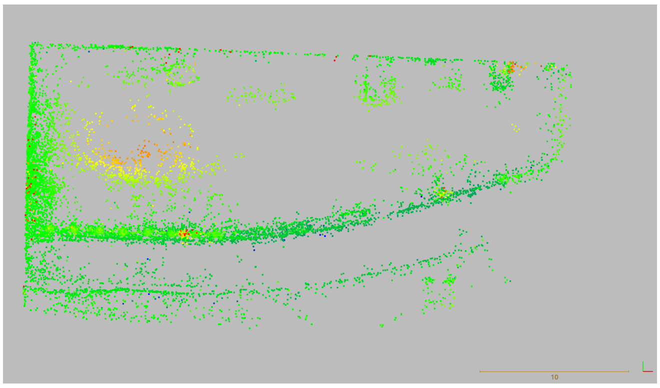


(c) Cumulative histogram of GT distance

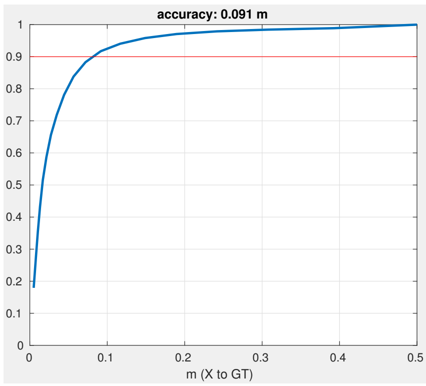
Figure 3: Sparse 3D map of the garden obtained from SfM reconstruction from the initial drives through the garden.



(a) Distance (m) of reconstructed point cloud section to the GT reference (top view)

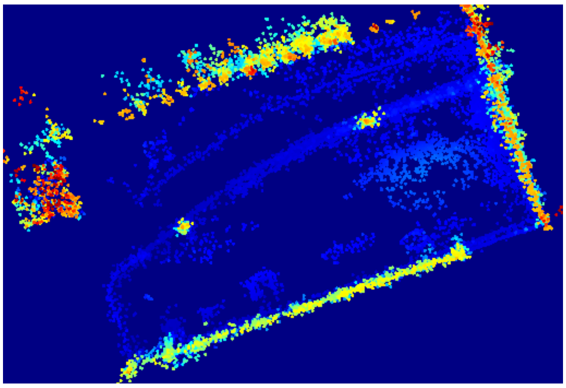


(b) Height-colored point cloud (top view)

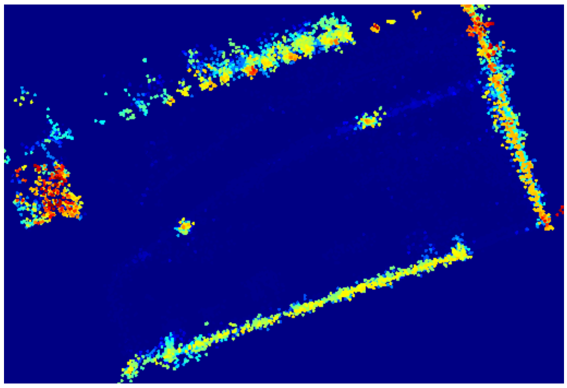


(c) Cumulative histogram of GT distance

Figure 4: Sparse 3D map of the garden, cropped to interior and 1m high section



(a) Height map

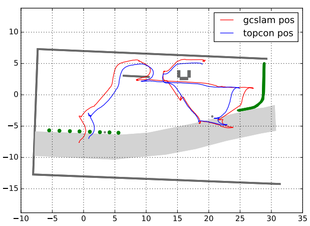


(b) Obstacle probability

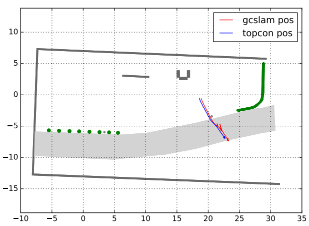
(c) Detected obstacles

(d) Combined with registered sketch map

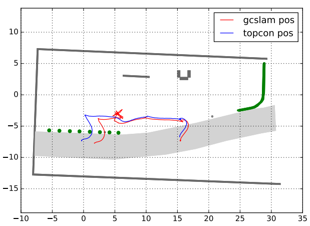
Figure 5: Occupancy map aligned to sketch map.



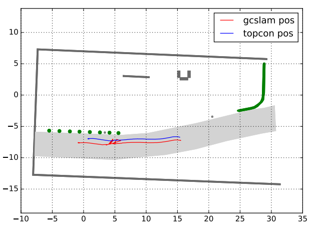
1) around garden



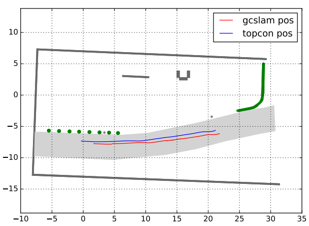
2) around lamp



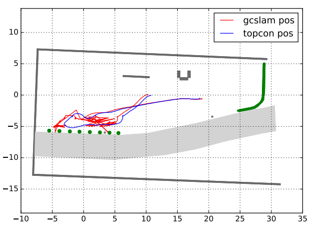
3) around obstacles



4) short distance on path



5) straight line on path



6) straight path on grass

Figure 6: Overview of recorded scenarios. Trajectories of Localization (red) and GT (blue), larger plots follow.

4.6 Localisation & Navigation

We have recorded several scenarios when the Demonstrator was autonomously navigating to different locations across the garden. The scenarios were captured after the demo event, in October 2017.

4.6.1 Recorded Data

The recorded evaluation dataset consists of 6 scenarios with total 13 minutes of 24 navigation plans executed shown in Figure 6. The following data were collected as standard ROS messages published during the execution and saved in bagfiles, in particular of the following types:

- **Goal** is the input 6D pose set by the user via the registered sketch map.
- **Plan** is an intended static path leading to the goal, computed by the planner while taking the static obstacle map into account. The plan is represented as a series of waypoints, ie. 2D poses (planar translation only) uniformly sampled in distance (0.1 m) ending at the goal position. When re-planning occurs, multiple plans can lead to the same goal, but the robot follows only the latest (active) plan. Goals active for < 5s, eg. when recovering from lost position, were omitted from the subsequent evaluation.
- **SLAM localisation** trajectory is a time-series of 6D poses (translation + rotation), ie. where the robot believes it is traveling at a given moment, based on the visual input from cameras. The pose is updated at $\sim$5 Hz rate, but drop-outs occur when a match with 3D map cannot be established (lost tracking). Navigation to follow the active plan is driven by the SLAM pose estimated wrt. 3D map coordinates.
- **GT trajectory** is a reference measured by a *Topcon* laser tracker (translation) and IMU (rotation), ie. where the robot precisely is at a given moment. The sampling rate is $\sim$12 Hz, short drop-outs occur when the robot gets out of the line of sight of the laser tracker, eg. behind the lamp for 1 s. The error related to GT measurement is given in Table 1.

Error	*Sensor*	*Registration*	*Total*
Translation (*Mean)*	2.9 cm (ATE*)*	6.3 cm	9.2 cm
Translation (*Max)*	16.6 cm (ATE*)*	12 cm	28.6 cm
Rotation (Mean)	0.2 deg	0.1 deg	0.3 deg

Table 1: Breakdown of GT evaluation accuracy. Comparison of GT to SLAM localisation is affected by sensor accuracy (laser tracker/IMU) and additional GT-to-SLAM-map registration accuracy (Sec. 4.5). *ATE*: Absolute Trajectory Error.

4.6.2 Observed Characteristics and Metrics

A number of characteristics was extracted from the recorded data for further analysis, illustrated on an example scenario. Plots for all scenarios can be found in Appendix 7.

- **Trajectory map** compares different trajectories shown in GT coordinates overlayed over the sketch map. Ideally, all three (GT/SLAM/Plan) should overlap. The SLAM trajectory and sketch map were transformed to GT coordinates using the estimated transformations described in Section 4.5.

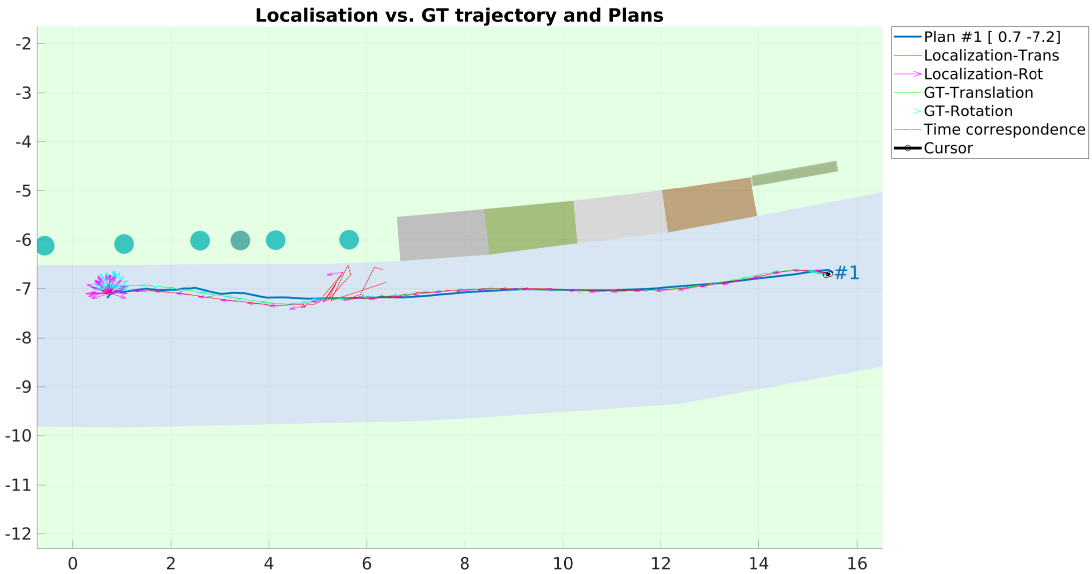


Figure 7: Trajectory map. Red lines in the center indicate a period of unstable localisation.

- **Position error** compares a given trajectory estimated by *SLAM* localisation to a reference GT trajectory, ie. measures the distance between locations corresponding to the same time, obtained with linear interpolation.

- **Cross-track error** (XTE) compares given trajectory (GT or estimated by *SLAM* localisation node) to a reference plan path, ie. shortest distances from all sample points on the trajectory to the active plan path, irrespective of time (Figure 11 right). Colors of plan plots correspond to trajectory map.

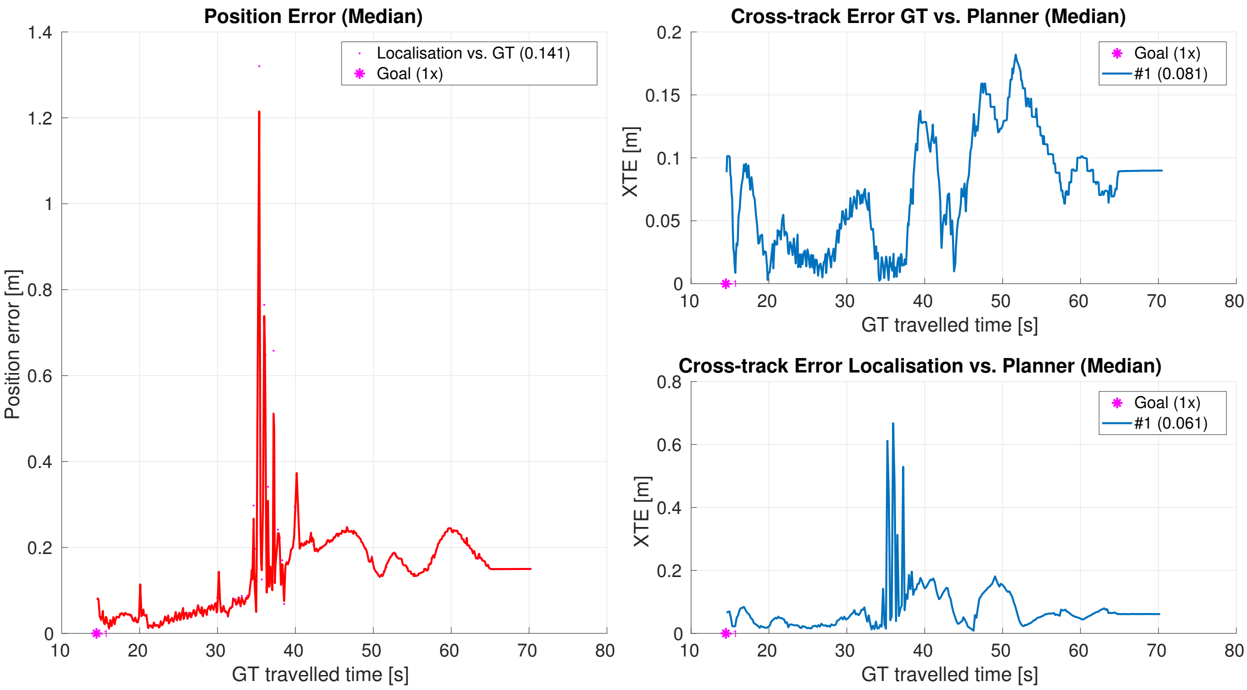


Figure 8: Position error plots. Median errors (m) and number of goals (x) are in brackets.

- **Relative position change** is computed as the Euclidean distance between two consecutive pose samples of a trajectory (GT or SLAM).
- **Position jump** occurs when the movement estimated by *SLAM* exceeds 2x the maximum GT movement. Its time series is filtered to indicate at most one occurrence within a 10 message long window.
- **Lost tracking** occurs when no pose is published for 0.5 s or the current position error exceeds 1.0 m.
- **Plan change** occurs when the active plan is replaced by a new one, ie. a new message arrives that leads to the same goal.

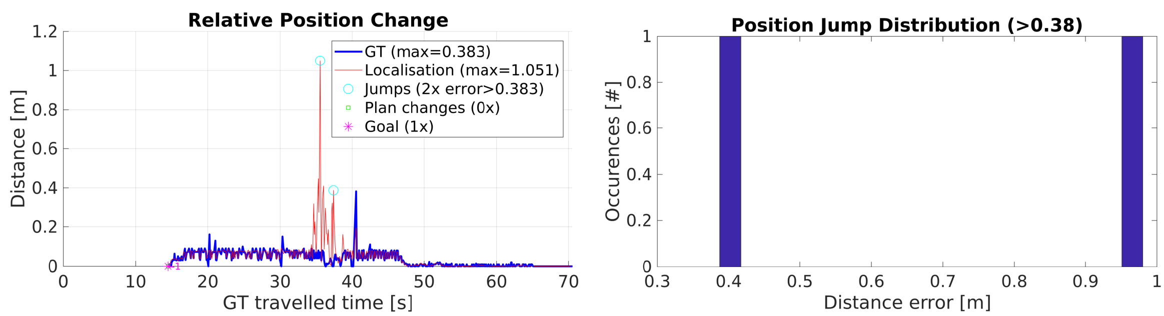


Figure 9: Plots for relative position change (left) and derived jump magnitude frequency (right).

- **Orientation error** compares given poses estimated by *SLAM* localisation to reference GT poses, ie. angle difference corresponding to the same time, obtained with linear interpolation, measured in two representations:
  - Euler angles: Yaw-Pitch-Roll (YPR) can be interpreted as Direction-Tilt-Rotation in camera context (Figure 11).
  - Minimum rotation angle that aligns the two poses (Min. axis-angle).

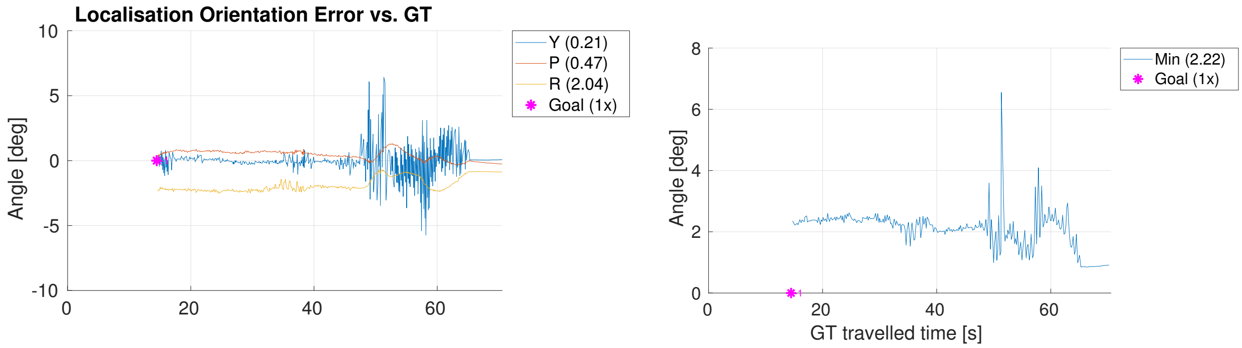


Figure 10: Plots of SLAM orientation error decomposed to YPR (left) and as minimum rotation angle (right).

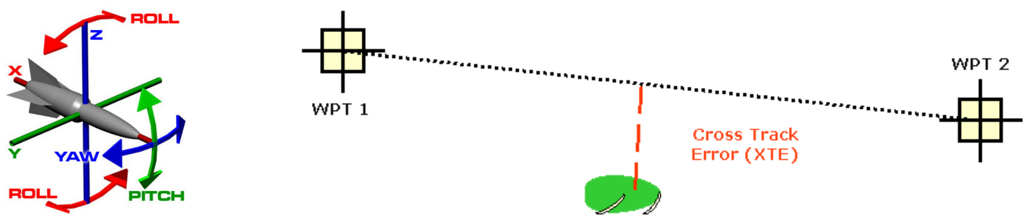


Figure 11: Schematic explanation of Yaw-Pitch-Roll (YPR) and Cross-Track-Error (XTE).

4.6.3 Summary Statistics

The evaluation of 6 scenarios is presented in Fig. 12, showing average characteristics described in the previous section. Median statistics were chosen to give estimates unbiased by jumps. The overall median values are given in Table 2 and compared to given limits, taking into account the accuracy of the used GT reference. Finally a t-test is performed to indicate the probability that the limit was not exceeded, ie. higher p-value is better.

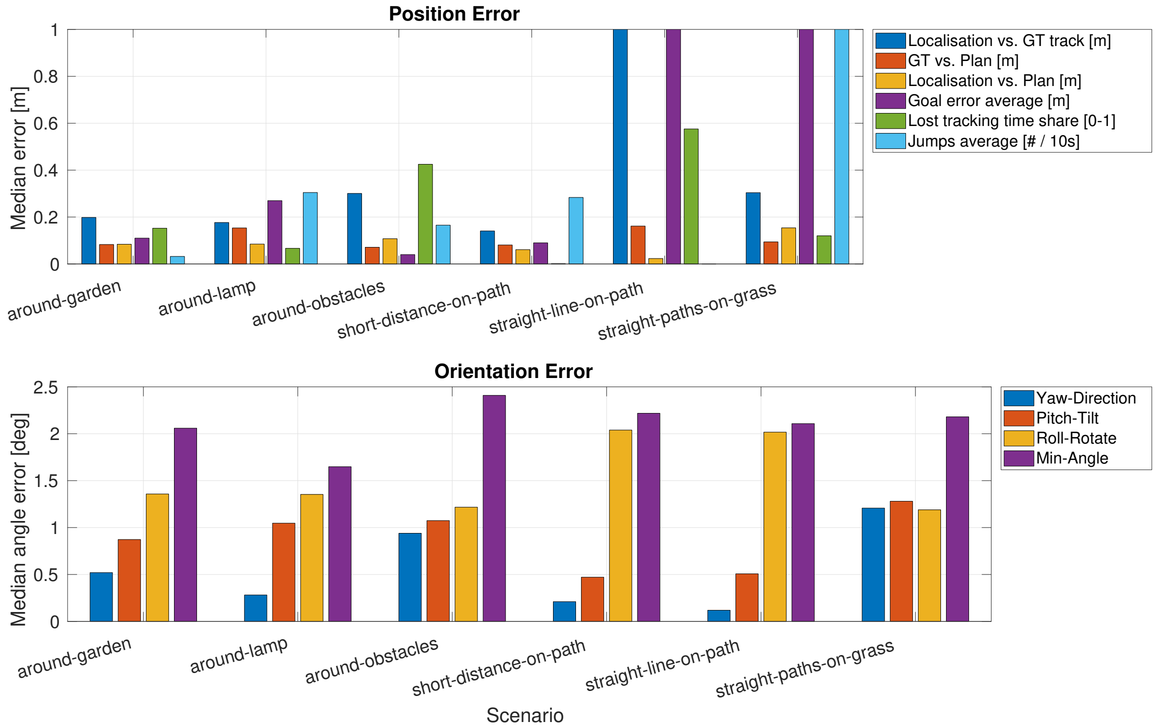


Figure 12: Scenario statistics for position, orientation and plan errors. Goal error average is calculated from position errors read at the end of the route segment leading to the goal. Lost tracking time share is calculated as the ratio of total lost tracking and total travel time.

Characteristic	*Value* $\pm \sigma$	*Limit*	*GT Acc.*	*Test*
Localisation vs. GT track (median error)	0.25 ± 0.30 m	0.10 m	0.09 m	p=0.008
GT vs. Plan error (median XTE)	0.09 ± 0.08 m	0.05 m	0.09 m	p=0.010
Localisation vs. Plan error (median XTE)	0.08 ± 0.04 m	0.05 m	0.01 m	p=0.000
Goal position error (median of succ. plans)	0.19 ± 0.10 m	0.20 m	0.09 m	p=0.704
Orientation error - Yaw (direction)	0.40 ± 0.25 deg	5.00 deg	0.30 deg	p=1.000
Orientation error - Pitch (tilt)	0.96 ± 0.37 deg	5.00 deg	0.30 deg	p=1.000
Orientation error - Roll (rotation)	1.36 ± 0.50 deg	5.00 deg	0.30 deg	p=1.000
Orientation error - Rotation angle (min)	2.14 ± 0.42 deg	5.00 deg	0.30 deg	p=1.000
Plan summary (ok / failed start / failed goal)	14 / 3 / 5	0		
Lost tracking time share	13.62 %	0.00 %		
Jumps average [# / 100s]	2.2×	0.0×		

Table 2: Evaluation summary across all scenarios (macro-average), including two-sample t-test for equal means (one-sided).

4.6.4 Discussion

The summary results given above in Table 2 suggest that the absolute accuracy of Localisation position does not meet the strict limits we imposed on it, while at the same time our ability to evaluate it is limited by the accuracy of GT measurements and alignment (Table 1). Plan following errors are lower than absolute because the position error accumulates more along straight plan segments rather than across and the former is not observed by XTE (except at path ends). The outcome for orientation is favorable, the limit was met with a margin.

The lost tracking and jumps distribution suggest that stability of the Visual SLAM estimation in time is limited, mainly due to to the fact that the current pose is estimated independently of the previous pose. We have analyzed the individual scenarios in Appendix 7 and discovered several problematic spots, where localisation repeatedly fails. In particular when navigating close to the row of bushes:

- `short distance on path, straight path on grass` - lost tracking when robot arrives close to the bushes,
- `straight line on path` - offset accumulates when approaching the bushes and ends with exactly two bushes to the left along the row,
- `around bushes` - the same location offset is kept while navigating in between the bushes, and the correct pose is recovered when the robot gets away from them.

We suspect the possible cause lies in the repetitive structure of surrounding building walls and bushes themselves. We will conduct systematic evaluation across the whole garden to discover patterns which lead to such errors. We expect this cause will not occur in the other project garden, which will be also evaluated.

5 Results

Functionality of individual components was assessed by practical trials covering their operational range based on random or predefined sequences of actions, following *D7.3 - Component and System Evaluation Plan*, and the above described demonstration procedure and evaluation.

Properties marked NA could not be tested yet because the corresponding component was not integrated or a feature not implemented, as mentioned in Section 3. These will be included in the upcoming *D3.2 - Implementation and evaluation of SLAM, 3D from binocular and motion stereo* and other deliverables.

5.1 Vehicle platform

Evaluated Characteristic	*Limit*	*Test*	*Result*
Random driving trials with expected payload			
- drives distance on grass, pavement in test garden	10 m	10 m	Pass
- drives up and down 1 m grass slope at inclination	10°	10°	Pass
- both forward and backward			Pass
- turning radius	1 m	0.5 m	Pass
Uninterrupted power supply to all system components	20 min	>20 min	Pass

5.2 Visual sensors

Evaluated Characteristic	*Limit*	*Test*	*Result*
Provides uninterrupted video stream			
- simultaneously 10 cameras	10	10	Pass
- composed field of view	360°	360°	Pass
- stereo coverage	270°	360°	Pass
- frame rate at WVGA resolution	5 fps	8 fps	Pass
- capture latency (max)	200 ms	18 ms	Pass
Objects in the range are in focus	0.1m - ∞	0.05 m - ∞	Pass
Handles back-light well		AEG	Pass
Handles changing outdoor light conditions well		AEG	Pass
Hot or cold pixels	few	none	Pass
Angle error of the camera arrangement	1°	1°	Pass

AEG: auto exposure and gain active

5.3 3D data processing and analysis

Evaluated Characteristic	*Limit*	*Test*	*Result*
Depth sensing range wrt. robot location	0.2 - 20m		NA
- error within 1 m of vehicle	10 mm		NA
- error within 3 m of vehicle	30 mm		NA
- relative depth error	10 %		NA
Sketch map registration (residual error)			
- ground surface control points error	50 mm	60 mm[1]	Fail
- map object locations error	50 mm	60 mm[1]	Fail
- map object shape parameters error	20 mm		NA
Drivable region localisation error	0.1 m		NA
- occupancy grid accuracy	50 mm	100 mm	Fail
Vehicle localization accuracy			
- error wrt. GT position	0.10 m	0.25 m	Fail
- error wrt. GT orientation	10°	2.1°	Pass

[1] translation error

5.4 Task specification, planning and execution

Evaluated Characteristic	*Limit*	*Test*	*Result*
Sketch map supports all garden objects			Pass
- click to location in the map publishes coordinates			Pass
- click at map object publishes garden object			Pass
Error of robot servoing to 0.1m from target plants			NA
Master controller coordinates tasks			Pass
Vehicle control actions correspond to the input			Pass
Random route following trials cross track error	0.05 m	0.09 m	Fail
Map-based garden navigation to a particular location			
- location accuracy	0.20 m	0.19 m	Pass
- location repeatability	0.20 m		NA
- orientation accuracy	10°	2.14°	Pass
- orientation repeatability	10°		NA
- mean time between failures (MTBF)	5 min	2 min 35 s	Fail

failure: collision, blockage or lost tracking

6 Conclusion

Based on the above given analysis, we can discuss how the given D7.2 demonstrator success criteria were met:

Vehicle sensor system can acquire accurate 3D data. The mean accuracy of 3D map was estimated to be 9 cm in the relevant part of the garden, ie. near bushes and ground. The first demonstrator was using only single sparse 3D method to acquire garden geometry (SLAM based on visual features) acting as a substitute to the dense methods (stereo), which will be integrated in the next steps to reconstruct shapes of garden objects.

Vehicle can navigate on grass and pavement. Successfully tested for both surfaces. They can be traversed in all directions. However, the used Platform 2 does not carry arm and effectors, which will add load to be carried by the next demonstrator vehicle Platform 3. It also has a different construction and will be tested independently.

Vehicle can estimate its location with a target accuracy of 5-10 cm. The achieved global mean accuracy was 25 cm, with the 9 cm tolerance of GT measurement we can conclude the 10 cm limit was not met but cannot confidently quantify the difference. Path following accuracy (cross-track error) was however 9 cm, which satisfies the upper limit. The higher global error was mainly due to the unstable periods when SLAM gets lost or the pose estimate jumps around. Following the proposal we will exploit additional sources of data (IMU, odometry, semantics) to constrain the estimated pose and improve temporal stability. Additionally, dense image analysis (scene flow) will be used to estimate the camera motion and fused with the other sources.

Vehicle can navigate to specified locations in the garden near hedges, and bushes without collisions. No collisions occurred, the localisation was accurate enough in the proximity of obstacles. The path planning took into account the robot size and error margin, which also prevented collisions. In open terrain the robot however diverged from the plan in some cases. For such situations the navigation algorithm will be improved to handle the localisation jumps better.

The evaluation was limited to the Bosch garden, which is challenging due to its sparsity (objects spread out), repetitive and reflective surroundings (walls, windows). We will continue with evaluation at the second project garden at Wageningen.

7 Appendix

This section presents statistics plots and trajectory maps for each evaluated scenario. Description of individual graphs can be found in Sec. 4.6.2, also see Sec. 4.6.4 for discussion.

7.1 Scenario `around-garden-2017-10-25-15-04-07`

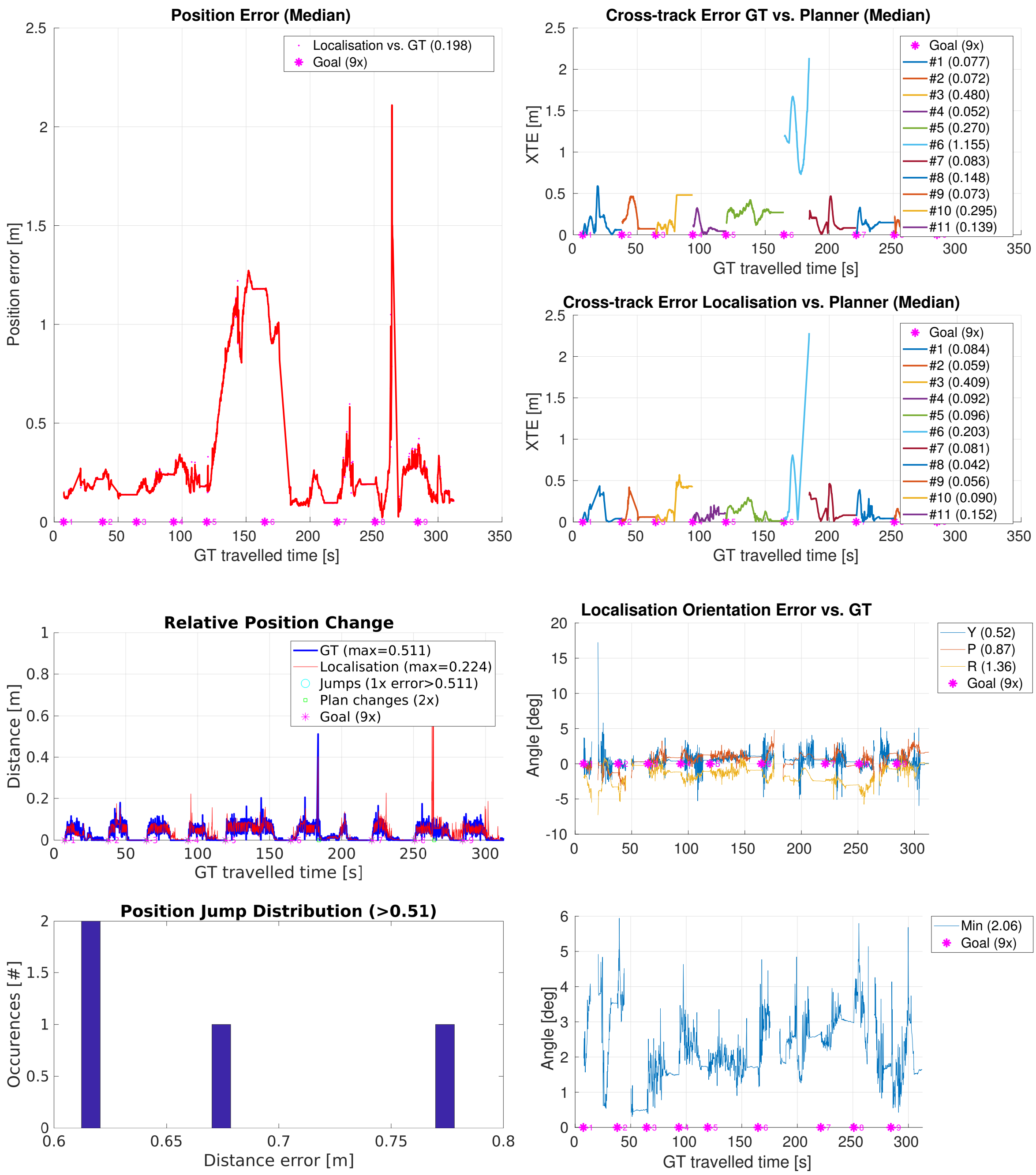


Figure 13: Statistics for `around-garden-2017-10-25-15-04-07`

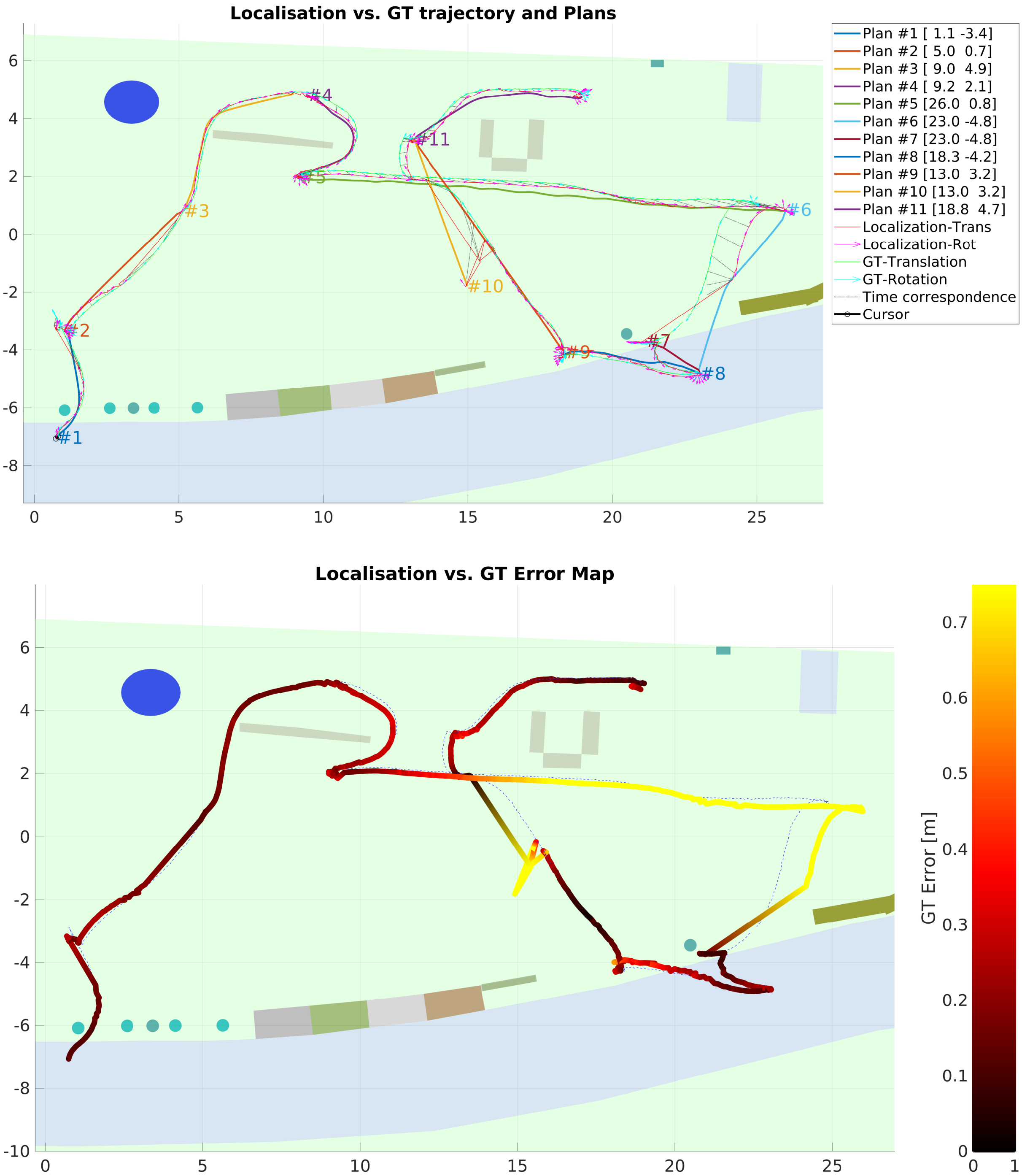


Figure 14: Trajectory maps for `around-garden-2017-10-25-15-04-07`

7.2 Scenario `around-lamp-2017-10-25-14-58-10`

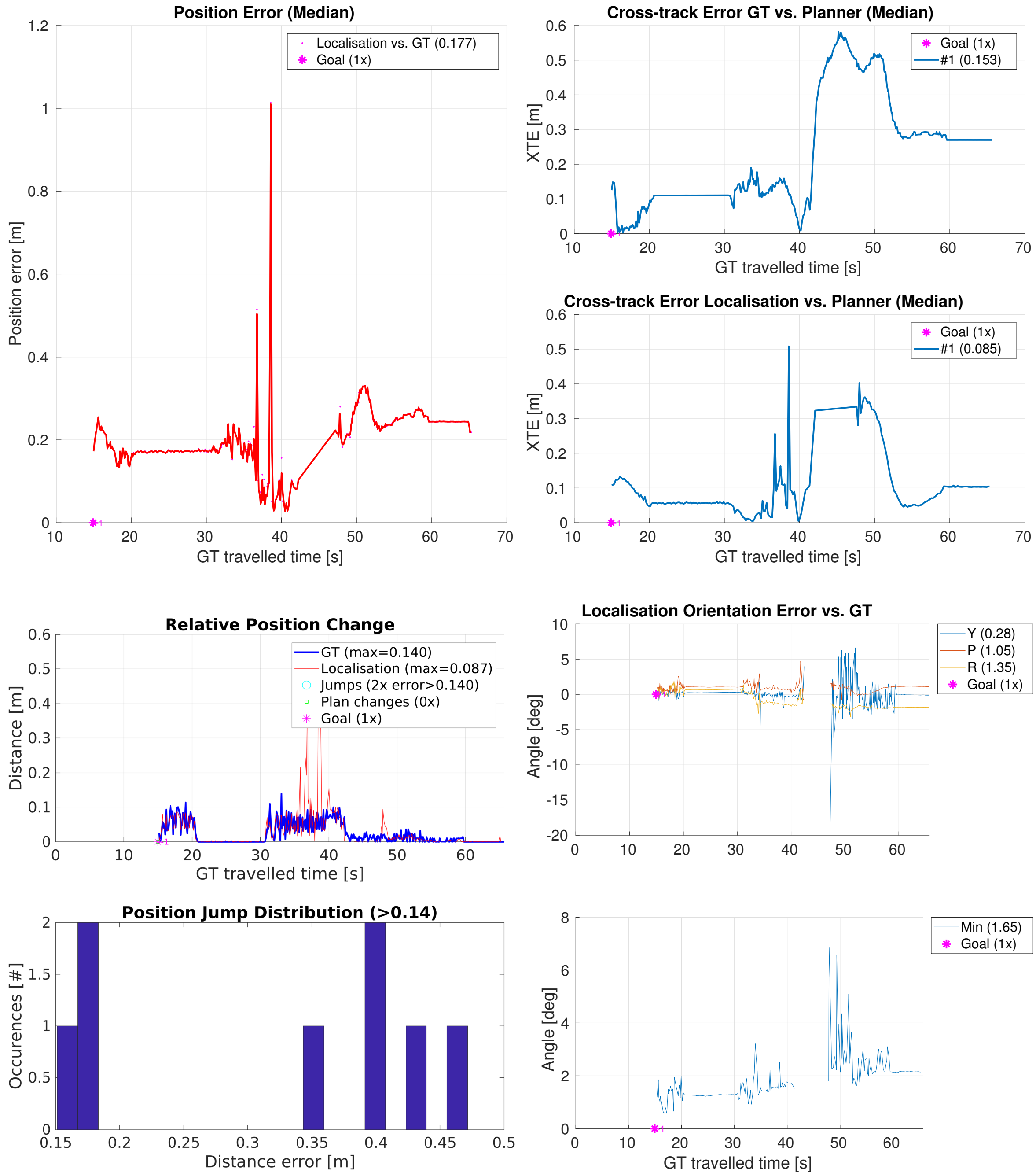


Figure 15: Statistics for `around-lamp-2017-10-25-14-58-10`

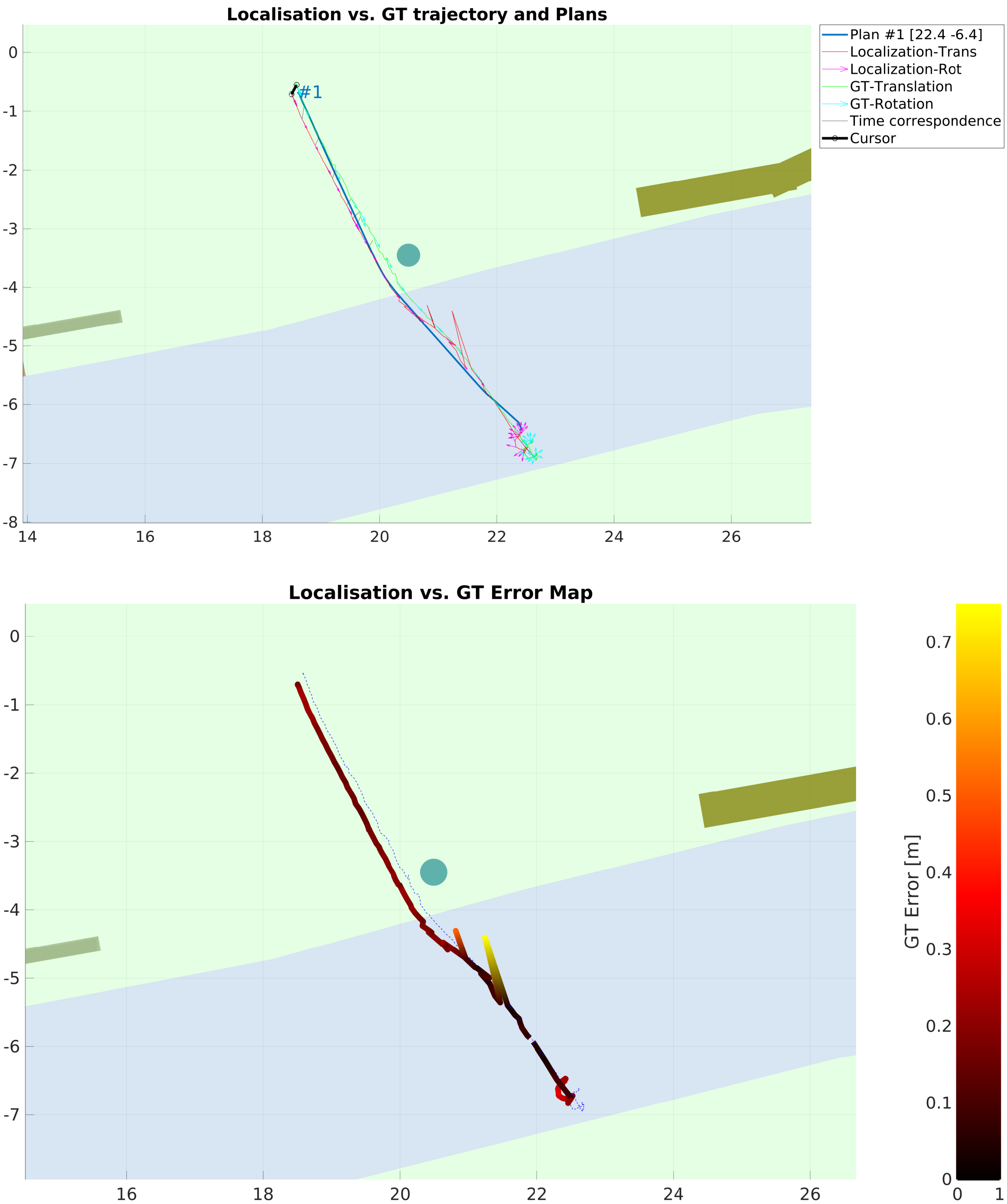


Figure 16: Trajectory maps for `around-lamp-2017-10-25-14-58-10`

7.3 Scenario `around-obstacles-2017-10-25-15-00-43`

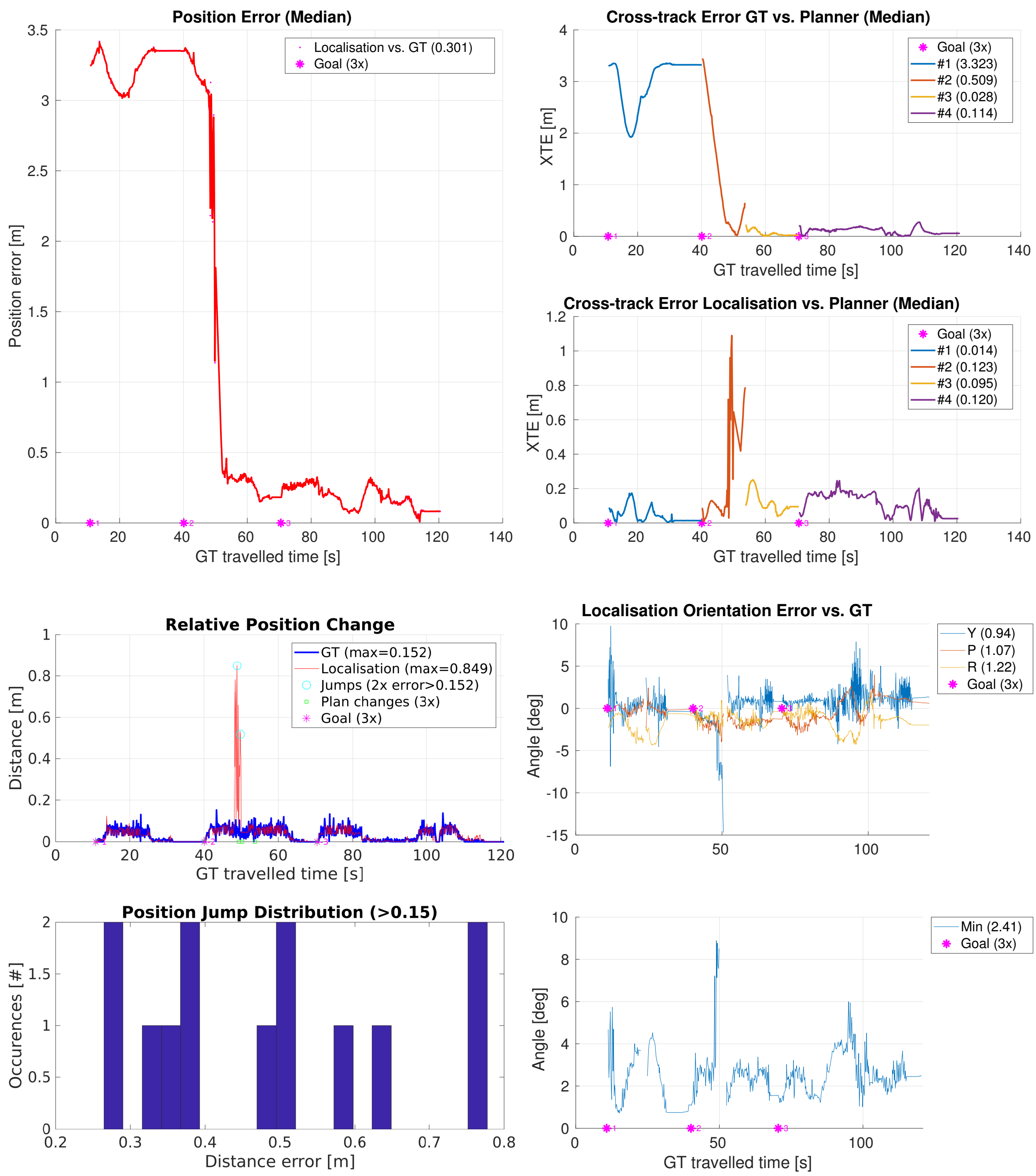


Figure 17: Statistics for `around-obstacles-2017-10-25-15-00-43`

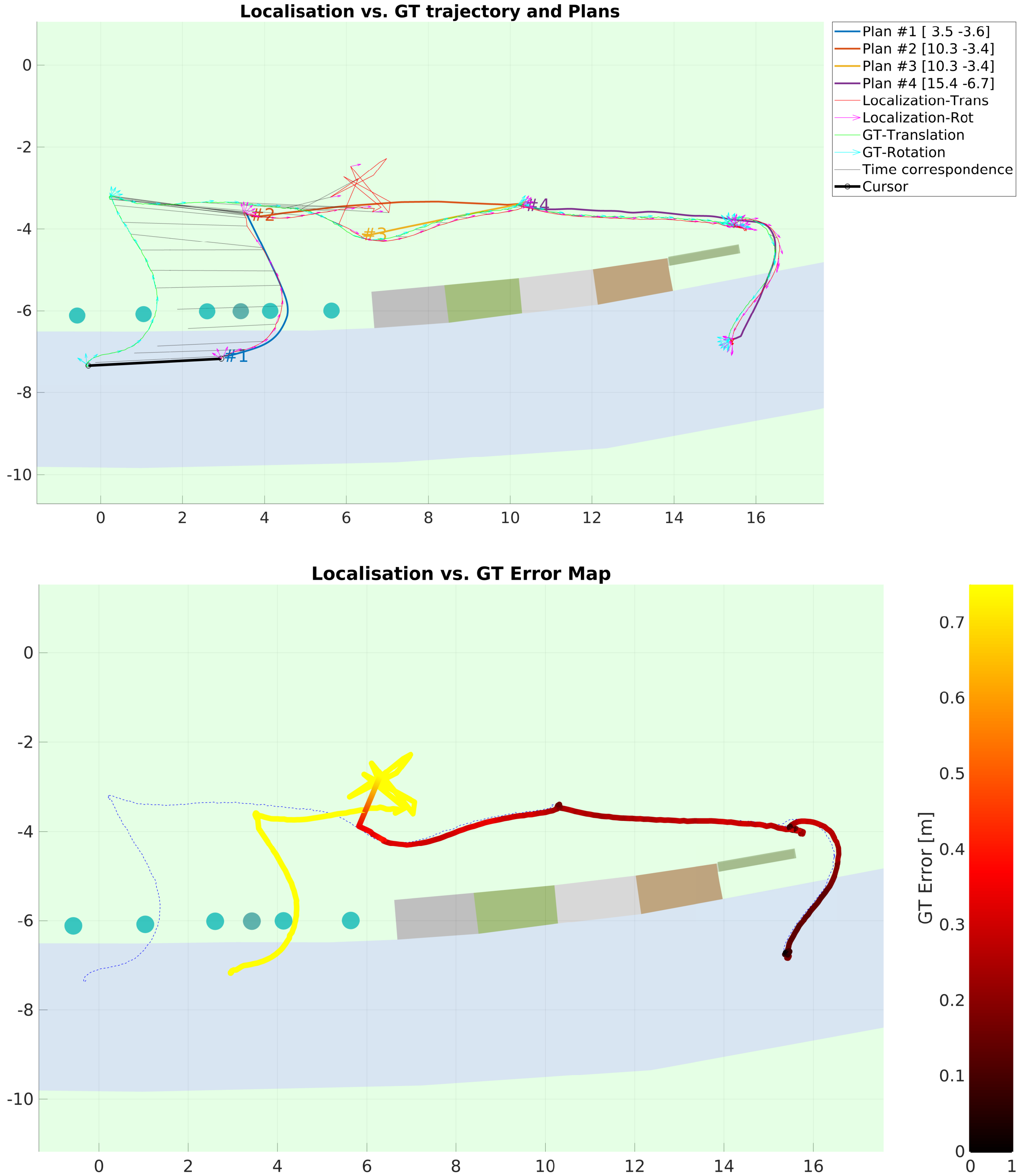


Figure 18: Trajectory maps for `around-obstacles-2017-10-25-15-00-43`

7.4 Scenario `short-distance-on-path-2017-10-25-15-02-48`

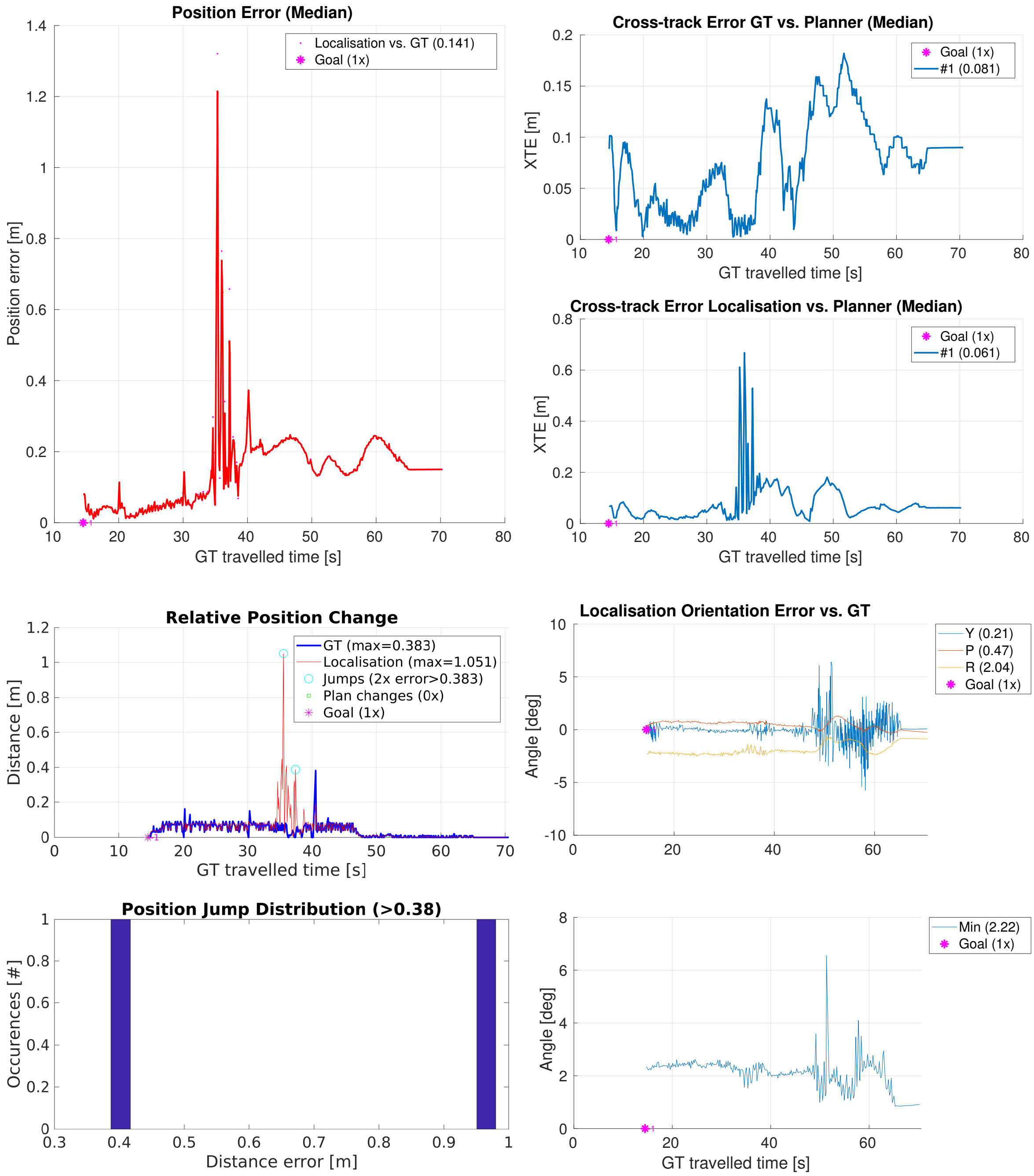


Figure 19: Statistics for `short-distance-on-path-2017-10-25-15-02-48`

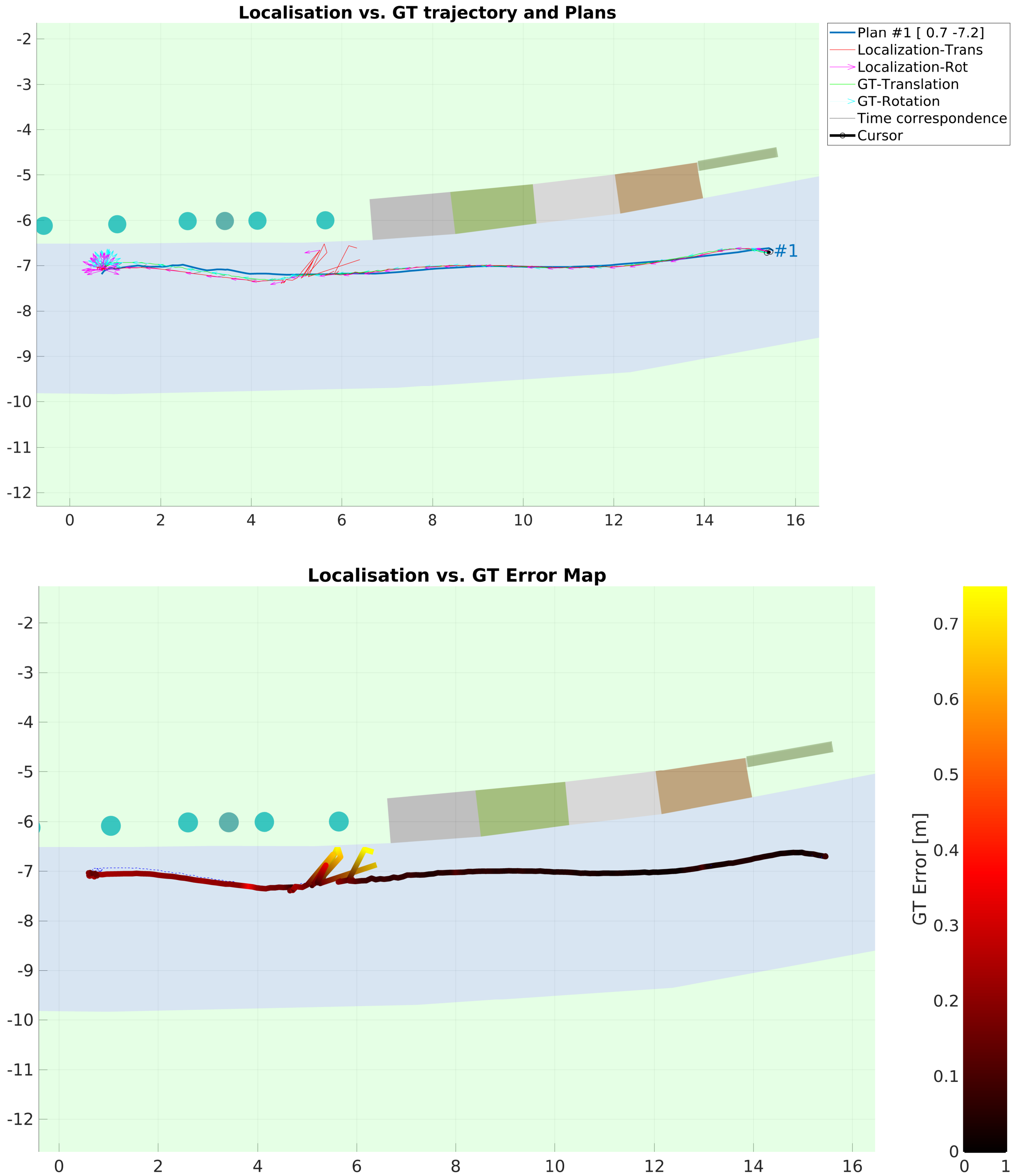


Figure 20: Trajectory maps for `short-distance-on-path-2017-10-25-15-02-48`

7.5 Scenario `straight-line-on-path-2017-10-25-14-59-29`

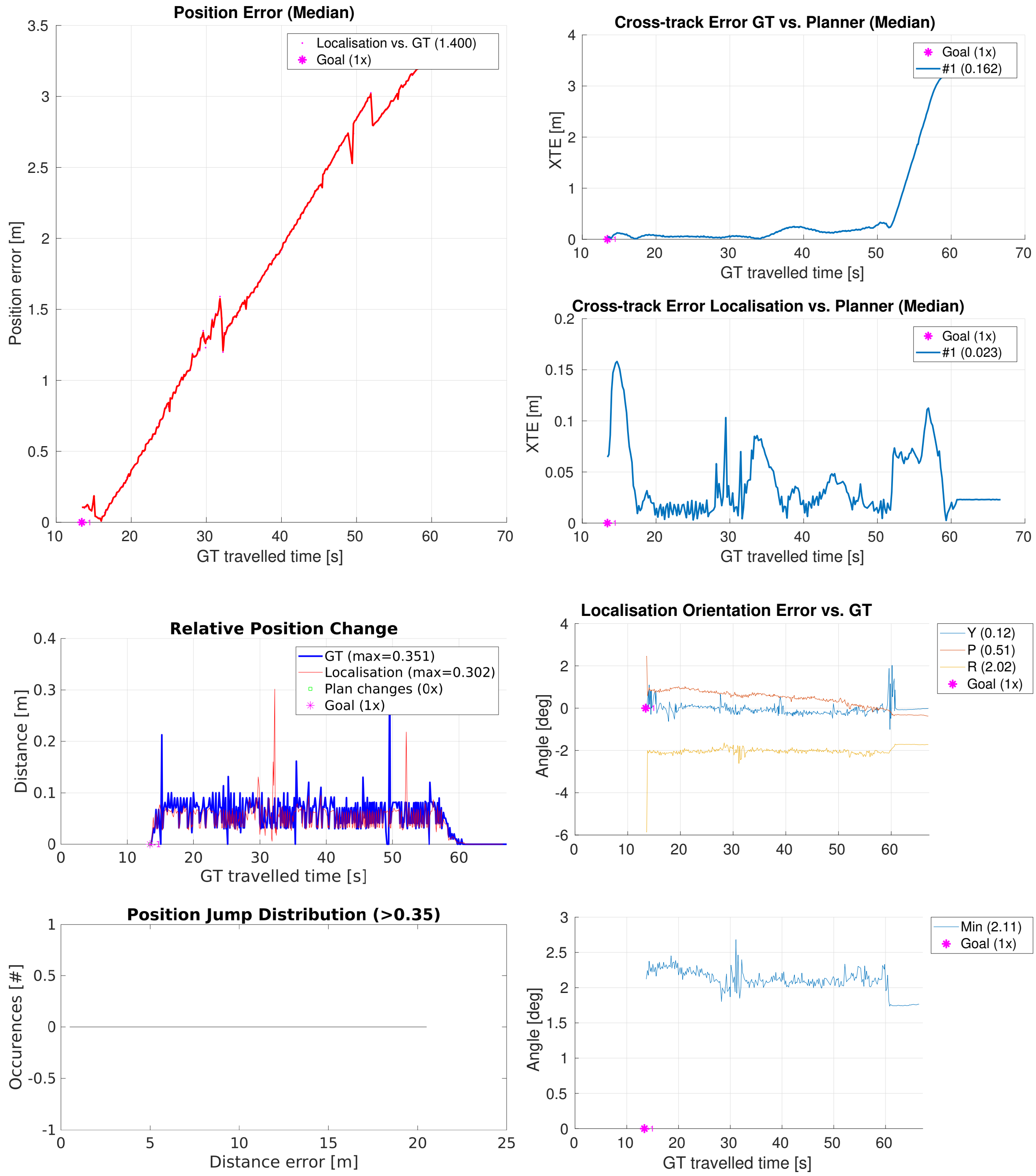


Figure 21: Statistics for `straight-line-on-path-2017-10-25-14-59-29`

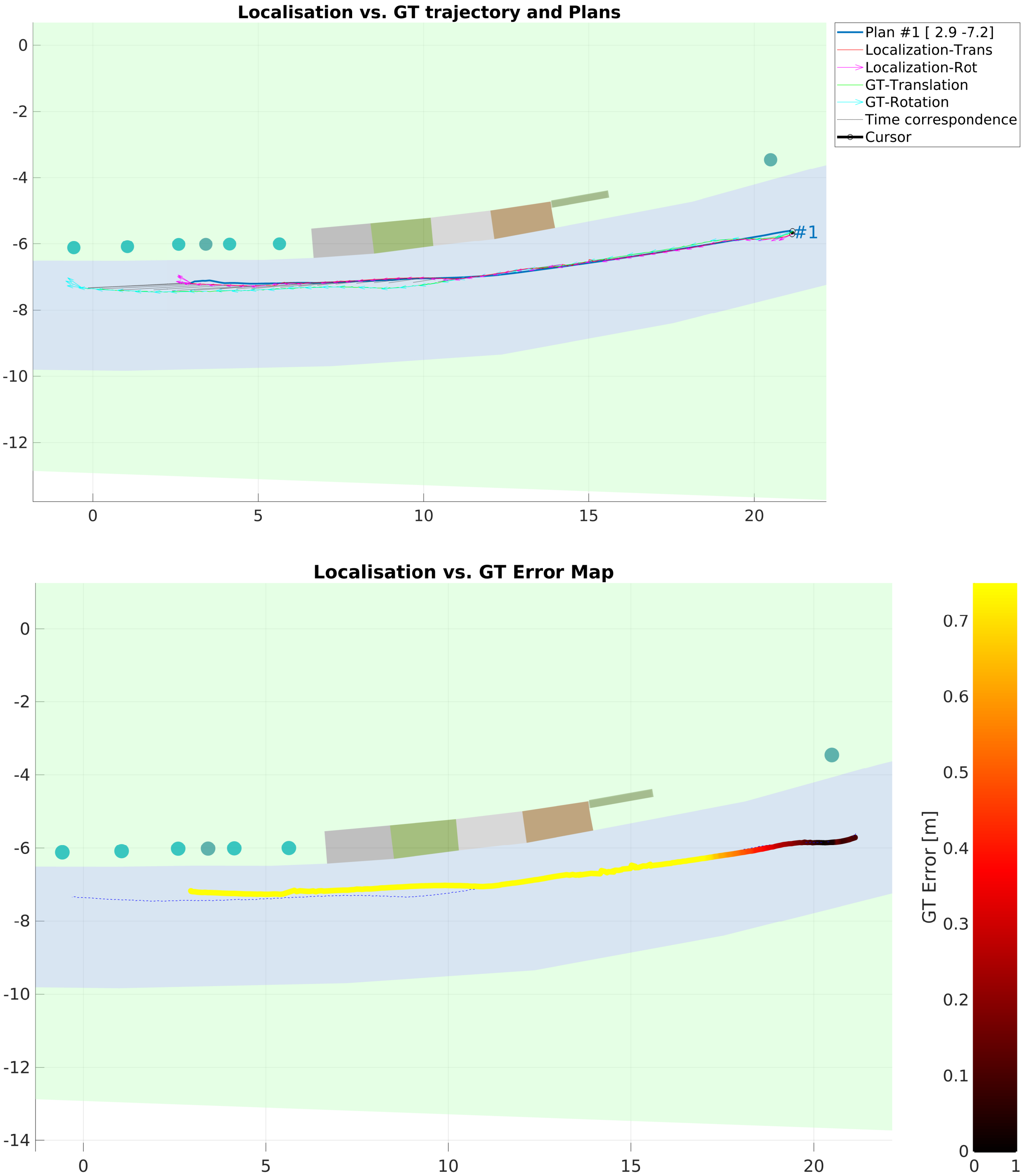


Figure 22: Trajectory maps for `straight-line-on-path-2017-10-25-14-59-29`

7.6 Scenario `straight-paths-on-grass-2017-10-25-14-54-53`

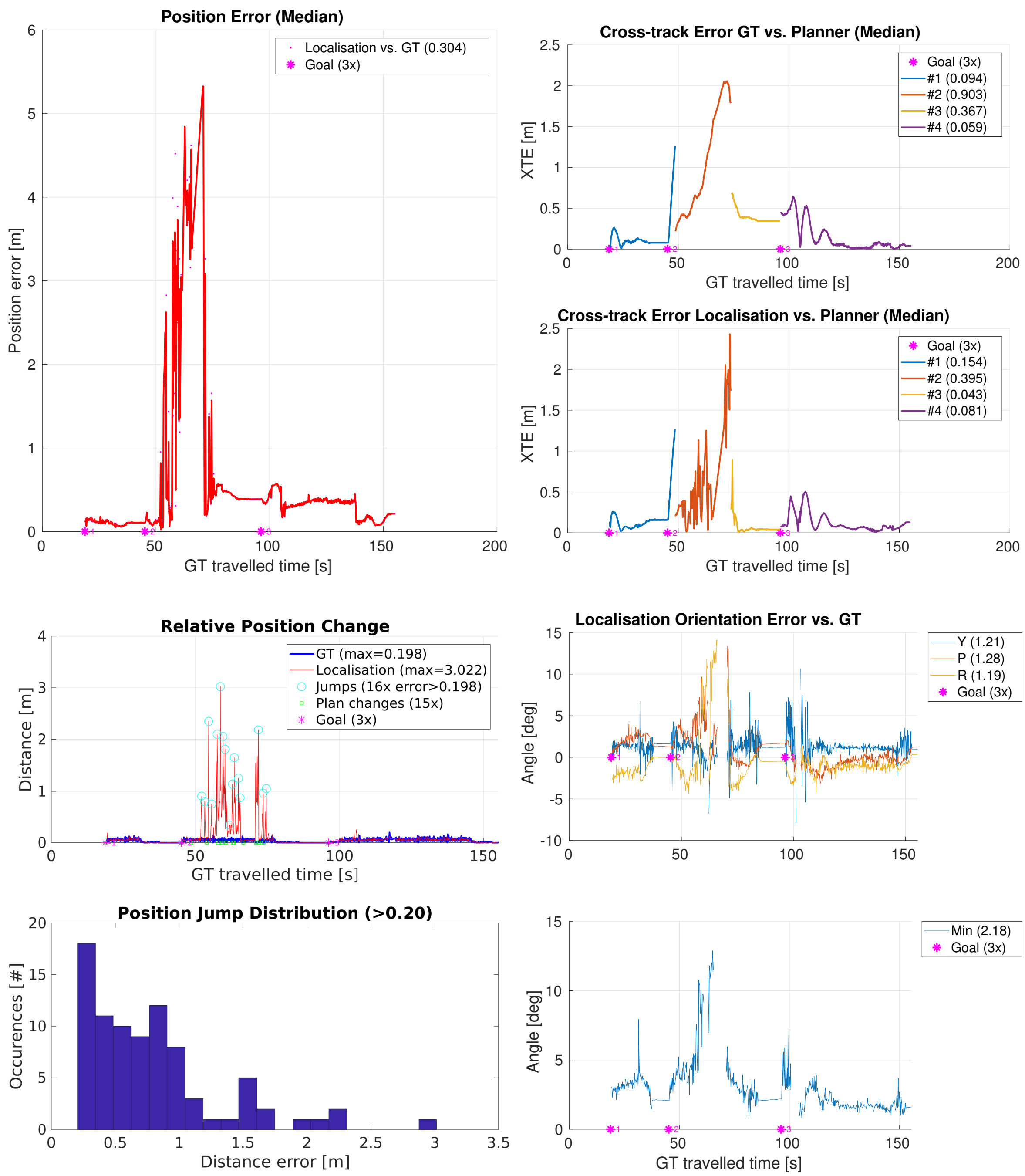


Figure 23: Statistics for `straight-paths-on-grass-2017-10-25-14-54-53`

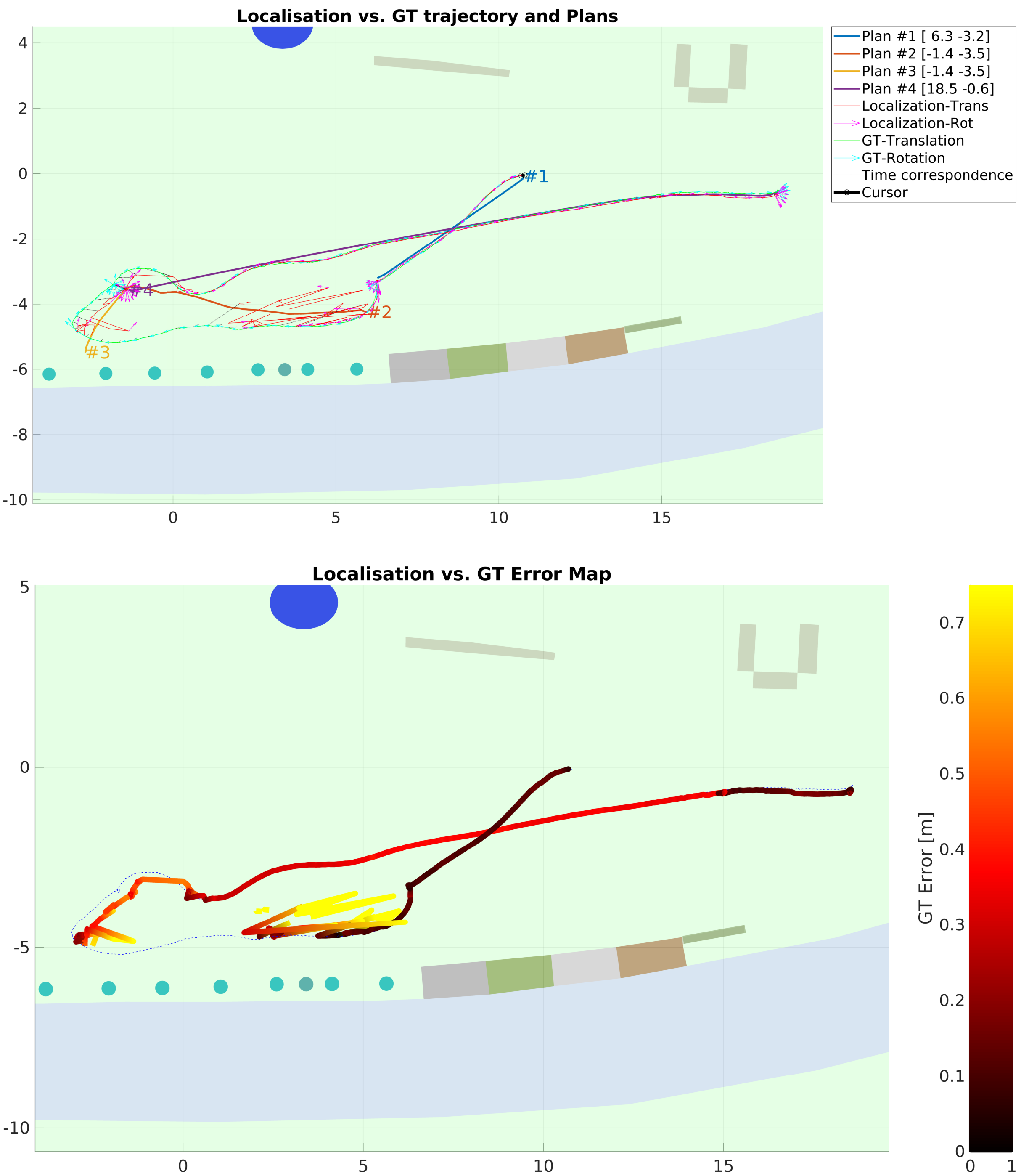


Figure 24: Trajectory maps for `straight-paths-on-grass-2017-10-25-14-54-53`

References

[1] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proc. CVPR*, 2017.

[2] Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. CVPR*, CVPR ’06, pages 519–528, Washington, DC, USA, 2006. IEEE Computer Society.

