

CREATING MULTI-TEMPORAL MAPS OF URBAN ENVIRONMENTS FOR IMPROVED LOCALIZATION OF AUTONOMOUS VEHICLES

J. Schachtschneider^{1*}, C. Brenner²

¹ Institute of Cartography and Geoinformatics, Leibniz Universität Hannover, Germany - julia.schachtschneider@ikg.uni-hannover.de

² Institute of Cartography and Geoinformatics, Leibniz Universität Hannover, Germany - claus.brenner@ikg.uni-hannover.de

KEY WORDS: LiDAR, Mobile Mapping, Dynamic Environments, 3D Modelling, Localization

ABSTRACT:

The development of automated and autonomous vehicles requires highly accurate long-term maps of the environment. Urban areas contain a large number of dynamic objects which change over time. Since a permanent observation of the environment is impossible and there will always be a first time visit of an unknown or changed area, a map of an urban environment needs to model such dynamics.

In this work, we use LiDAR point clouds from a large long term measurement campaign to investigate temporal changes. The data set was recorded along a 20 km route in Hannover, Germany with a Mobile Mapping System over a period of one year in bi-weekly measurements. The data set covers a variety of different urban objects and areas, weather conditions and seasons. Based on this data set, we show how scene and seasonal effects influence the measurement likelihood, and that multi-temporal maps lead to the best positioning results.

1. INTRODUCTION

Nowadays there is a strong trend towards automated or even autonomous driving. One important element for the development of autonomous vehicles are very precise and up-to-date models or maps of the environment. As natural environments change over time, it is virtually impossible to provide such up-to-date maps. In the past, a frequent solution to this problem has been to identify static parts of the environment, which will then be the only parts that will be included in the map (Thrun, 2002). However, the urban environment consists of a variety of objects that behave differently. While there are indeed static objects such as buildings or the road surface, which are unlikely to change, other objects like vegetation will behave periodically throughout the seasons. Parked vehicles and other mobile objects may change their position in a daily or weekly cycle, let alone moving objects such as pedestrians and cars, which change within seconds (Cheng and Sester (2018), Bock et al. (2016)). Therefore, the removal of all objects which are non-static will also dismiss a large amount of possibly useful information. While probabilistic mapping methods assume that contradictions in the data are caused by sensor noise, there are very few approaches which explicitly model changes of the environment (Thrun et al., 2006).

Other approaches model defined states of dynamic objects, for example open and closed doors (Stachniss, 2009). This reduces the complexity of the environment model, but also is limited to a fixed set of states and can not adapt to unexpected changes, which makes such models unsuitable for uncontrolled outdoor environments.

Meyer-Delius et al. (2012) apply Hidden Markov Models (HMM) to occupancy grids of dynamic environments. They assign to each cell a probability about the occupancy state plus the state transition probability of the HMM. For validation they used two different settings: one is a parking lot that was scanned

twelve times during a day by a SICK LMS laser range finder. The second test scenario is a small hall, where people are crossing from an office to two different exits.

Another explicit model of dynamic behavior is the spatio-temporal environment model presented by Krajník et al. (2015). It uses the spectral domain to model temporal changes and also provides an exploration strategy to acquire measurements in different locations on a reasonable basis. Krajník et al. (2015) use occupancy grids, topological maps and feature-based maps as environment models and test their approach on two real world data sets each collected over a period of several months.

Biber et al. (2005) define three requirements for long term maps: 1. The lifetime of a map shall not influence the time that it needs to adapt to a change. 2. The map needs to be robust to outliers. 3. The map shall not interpolate between measurements. Their approach does not explicitly model dynamic parts of the environment, but covers those by multiple temporal maps with different timescales. They test their map learning system in an indoor environment performing three runs per day over a period of five weeks.

2. APPROACH

In this paper, we investigate the influence of scene content and scene changes on vehicle localization. Usually, robotic localization is based on a probabilistic approach where, at each time step, a belief is computed, which is a maximum a posteriori estimation based on a prior and the measurement likelihood (Thrun et al., 2006). The measurement likelihood in turn is usually modelled by a function which assesses the degree of agreement between the actual measurement data and the corresponding predicted data, obtained from an assumed robot pose and a map (this function often being the Gaussian density). Under ideal circumstances, a correct robot pose leads to a high likelihood, and ideally, there will be a sharp peak around the correct pose.

* Corresponding author

Temporal changes lead to large (as compared to the measurement accuracy) discrepancies between real and expected measurements. If this is not considered in the measurement model, it results in much too small likelihoods. In contrast, if it is modelled as outliers, the corresponding measurements are discarded. In highly dynamic scenes, this may result in large parts of the measurement data becoming unusable, which means that the pose estimation becomes brittle since it is based on very little data. Therefore, the goal of multi-temporal maps is to predict, for any given point in time, the environment in such a way that discrepancies between real and expected measurements are minimized and, since they are inliers then, a large part of these measurements can be used for pose estimation. In an informal way, one could say that robots that have the most accurate idea of their environment are the least surprised about their measurements and, therefore, can act in the most confident way.

Apart from temporal changes, object characteristics influence the likelihood function. Measurements to rigid objects, like walls, will lead to highly repeatable measurement values, which vary just by the precision of the measurement system, whereas vegetation, e.g. hedges, will lead to a wider spread.

In this paper, we investigate both effects. To this end, we have undertaken a major measurement campaign over a time span of one year. We aligned the data with a high precision so that our results are not influenced by systematic offsets from different measurement epochs. The result is a map for each epoch, and a map comprising all epochs. We then assess the influence of the scene content and dynamic scene changes on the computation of the likelihood function, in our case, concentrating on the deviations between true and predicted measurements and the number of inliers.

Using this setting, we evaluate the following three hypotheses:

1. The maps are suitable for localization. The true location leads to the smallest deviation and most inliers. Scenes containing vegetation lead to a spread of the likelihood function.
2. The maps reflect seasonal effects. Taking a map based on an epoch with a small temporal distance leads to a better performance.
3. If several epochs are used to create the map, the result improves significantly.

3. DATA

For this work we use data from a long-term measurement campaign. Starting in March 2017, we performed measurements with a LiDAR Mobile Mapping System about every two weeks along a 20 km route through Hannover, Germany (see Fig. 1b). The measurement area includes inner city areas as well as residential districts in suburbs, multi-lane roads, various intersections, parking lots, tram lines and areas with high cycling and pedestrian traffic. Due to the long measurement period, the data set contains different seasons with various weather and lighting conditions as shown in Fig. 2.

The data was acquired by a Riegl VMX-250 Mobile Mapping System (MMS) RIEGL VMX-250 (2012) which is mounted on a measurement vehicle as shown in Fig. 1a. The MMS contains two Riegl VQ-250 laser scanners which have a maximum

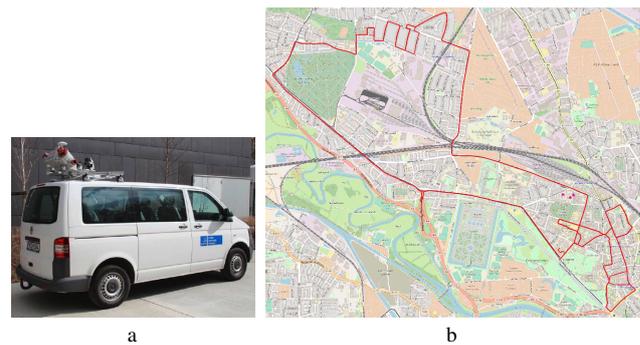


Figure 1. (a) Measurement vehicle with VMX-250 mounted on the roof top, (b) Route of the biweekly measurement campaign



Figure 2. Example images of the same scene in different seasons (a): 17-03-28, (b): 17-04-28, (c): 17-06-06, (d): 17-06-20, (e): 17-08-08, (f): 17-10-04, (g): 17-11-07, (h): 18-01-17, (i): 18-02-15; the change in the vegetation is most notable.

overall scanrate of 300,000 points per second with a ranging accuracy of ten millimeters. In addition, the system contains a localization unit and four cameras.

The highly accurate GNSS/INS system is combined with a Distance Measurement Instrument (DMI) for localization. The trajectories are obtained by a post-processing using reference data from the Satellite Positioning Service (SAPOS). This leads to an overall accuracy of the trajectory in the decimeter range.

For this work we use a subset of 14 measurement epochs which took place from March to October 2017 (see Fig. 3). The point clouds of those epochs were aligned using the strip adjustment approach from Brenner (2016). The result has a standard deviation below two centimeters.

Our work is based on laser point clouds; for the described period we have over 5 billion laser points in total. Fig. 4 presents the high quality and density of our data set. It shows an overlay of all epochs, where each epoch is colored using the

color scheme shown in Fig. 3. It can be seen that static objects, such as facades, show a dense mixture of all colors, except for areas where facades are occluded by vegetation. Tree crowns, on the other hand, show many small regions of distinguishable colors, corresponding to the extents of the crown at different growth periods. Pedestrians (at the bottom of the figure) are single colored since they are present in a single epoch only.

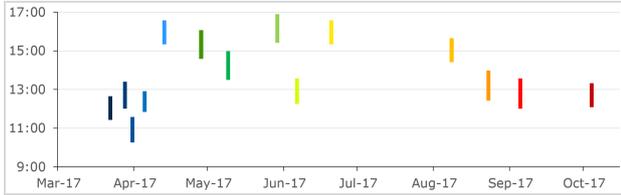


Figure 3. Temporal distribution of the 14 measurement epochs used for this paper; Bars colored by measurement epoch (blue = oldest, red = newest measurement)

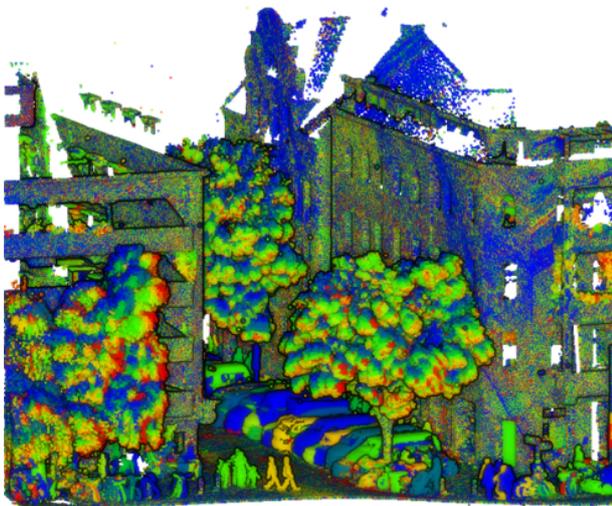


Figure 4. Example point cloud, colored by epoch (blue = oldest, red = newest measurement)

4. MAPS AND TEST SETS

For our experiments we divided the data set into epochs for modelling and testing. We use the point cloud from measurement epoch nine (17-06-20) as a test set *Test E9*, i.e. data of this epoch is used as true measurements and will be compared with different maps, created from the rest of the data. We compare four different maps: three maps from different seasons, which were created from one single measurement epoch each (*Map 1*, *Map 8*, *Map 14*) and one map created from all measurement epochs (*Map All*) (for details, see Fig. 5 and tab. 1). We selected the different epochs in a way that one is temporarily close to the test epoch (*Map 8* - 2 weeks), one is in spring (*Map 1* - three months earlier), and the third one in October *Map 14*, i.e. 4 months later. In this way, the temporal effects of the vegetation will be present in the data. We use a grid based map representation and sort all points into a voxel grid with a voxel edge length of two centimeters. Voxels which contain at least one reflecting point are marked as occupied.



Figure 5. Input data for maps (*Map E1*, *Map E8*, *Map E14*: green; *Map All*: green and grey) and test set (*Test E8*: red)

In order to evaluate the likelihood, we extracted trajectory snippets of about eight meter length from our test data and sub-sampled the corresponding point clouds to a point distance of maximum ten centimeters. Fig. 6 shows the example test areas. Bounding box *BB 1* is an area with mostly static parts, such as facades, road surface and sidewalks (see Fig. 6a). Fig. 6b shows bounding box *BB 2*, which additionally contains a tree and is used to analyze seasonal effects.

Name	Epochs
<i>Test E9</i>	17-06-20
<i>Map E1</i>	17-03-22
<i>Map E8</i>	17-06-06
<i>Map E14</i>	17-10-04
<i>Map All</i>	17-03-22 17-03-28 17-03-31 17-04-05 17-04-13 17-04-28 17-05-09 17-06-06 17-07-04 17-08-08 17-08-23 17-09-05 17-10-04

Table 1. Measurement epochs used for maps and test set

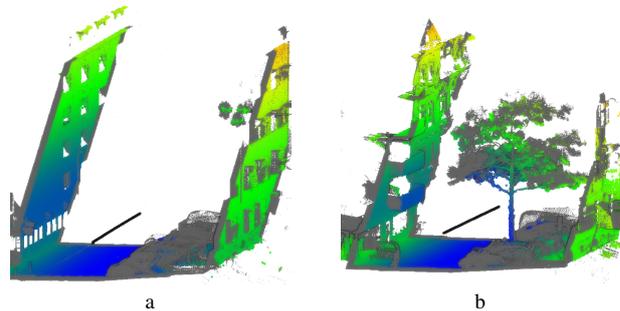


Figure 6. Example test sets *BB 1* (a) and *BB 2* (b) with *Map All* in grey; Points of the test sets are colored by the measured distance (short measurements = blue, far measurements = red). The black line above the road surface is the trajectory of the vehicle.

5. EVALUATION OF DIFFERENT MAPS

To obtain the response of the likelihood, the trajectories of the test areas *BB 1* and *BB 2* were shifted systematically in a seven times seven grid of one centimeter edge length. This produces 49 subsets for each test trajectory.

For each of the shifts, we assess the likelihood as follows. Using a ray tracing algorithm (Amanatides and Woo, 1987), we find, for each measured point, the closest occupied voxel in the map which lies along the scan ray. We then compute the discrepancy d_v between the voxel and the measured point, which is treated as a signed distance since the intersection of the ray with the voxel may occur before or after the real measurement. Fig. 7 illustrates the ray tracing approach for the case where the closest voxel on the ray lies in front of the actual measurement.

Instead of computing a single likelihood value from all the rays, we collect all d_v in a histogram of discrepancies. If the discrepancies are within the tolerance range of -0.06 m to $+0.06$ m, they are considered being inliers (in this way we take the high accuracy of the map into account). Apart from the histogram of discrepancies, we also calculate the percentage of inliers.

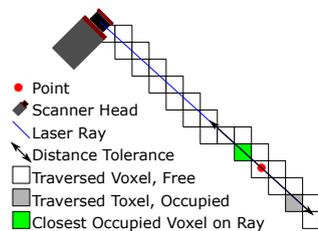


Figure 7. Illustration of tracing a LiDAR beam to determine the occupied voxel in the map which is closest to the measured point, within the distance tolerance range.

6. RESULTS AND DISCUSSION

In order to prove our hypotheses from section 2, we will discuss some of the histograms of discrepancies d_v . Note that the histograms in figs. 8–13 are normalized. As noted above, all points with a d_v within the range from -0.06 m to $+0.06$ m are counted as inliers.

To confirm hypothesis 1, we first examine the change in distances when the robot pose is modified. For the case of *BB 1*, the histograms in Fig. 8 and 9 show that when the robot position deviates from the correct position at $x = 0, y = 0$, the percentage of points with a small absolute distance $|d_v|$ decreases (blue bar in the middle of the figure goes down from 60% to 45%, 23% and 20%). Taking a close look at Fig. 8, one can see that secondary local maxima appear in the histograms at the positive and negative shift distance, as expected. The effect is also illustrated in 3D space, for a selected shift of $x = 3$ cm, in Fig. 10. As expected, the shift leads to positive discrepancies (yellow-red colors) on one facade and negative discrepancies (green-blue colors) on the opposite facade. Note that since the facades in our maps are almost parallel to the y -axis of our local coordinate system for this example, the shift in x -direction has a larger influence than the shift in y -direction, which is why the effect does not appear in Fig. 9. (Clearly, since the trajectories were shifted in the x - y -plane, the shift has also no influence on points on the ground.)

The same experiment for *BB 2* yields slightly different results, see Fig. 11. As can be seen, the percentage of measurements which fits the expectations (blue bar in the center) also decreases with increasing robot displacement. However, the decrease is not as strong as in the previous case. Here, we observe the spread in the likelihood induced by the presence of vegetation, as predicted in our first hypothesis.

With regard to the effects of temporal distance, Figures 12 and 13 show the distance histograms of *BB 1* and *BB 2* with the maps *Map E1*, *Map E8*, *Map E14* and *Map All*. They confirm hypothesis 2: for both bounding boxes, the seasonal map *Map E8*, which is the closest one in time to the test set, fits best, i.e.

has the largest percentage of inliers (not considering plot d, i.e. *Map All*, for now). In *BB 1* the effect is very small. Since it does not contain any vegetation – this was expected. Figures 14 to 17 show the point clouds of *BB 2* compared to the different maps. The voxels of the maps are colored grey and the points of the test set are colored by the discrepancy d_v . Points with a low d_v are green, here the found voxels are very close to the measured points, i.e. fall into the same voxel. If the voxel lies in front of the point, the point is colored green - blue and if the voxel lies behind the point, the point is yellow - red. The most points with a relatively large d_v are in the treetop. Due to seasonal growth of the foliage, here is the biggest difference between the different epochs. *Map E8* and *Map All* contain the most inliers. This is expected, because *Map E8* is closest in time to the test set and *Map All* was merged from all epochs, so it also contains the points from *Map E8* and additionally all other epochs.

Hypothesis 3 is definitely confirmed. *Map All*, containing all measurement epochs clearly has the smallest distances between points and corresponding voxels for all bounding boxes, see Fig. 12 and 13. For 90% (*BB 1*) and 80% (*BB 2*) of all measurements there is a corresponding voxel found. As shown in Fig. 18 and 19 it is also most robust against the shift, as a wrong position is still rated as good. This, however, also shows the disadvantage of adding all measurements to a map without any update concept: The map fills up with points which do not exist any more, like the foliage in the tree crown which is much denser in *Map All* than in the others or the parked vehicles, which overlap. This leads to conflicts with the current state of the environment. For a point in a parked car, a voxel is found in the map, which belongs to a completely different car.

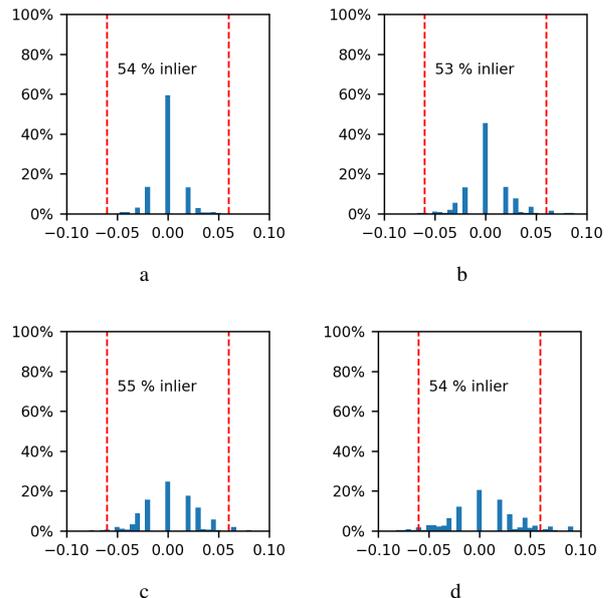


Figure 8. Distance histograms of bounding box *BB 1*: *Test E9* in *Map E8*, shift- $y = 0$, (a) shift- $x = 0$, (b) shift- $x = 1$ cm, (c) shift- $x = 2$ cm, (d) shift- $x = 3$ cm (x -axis: d_v in meters)

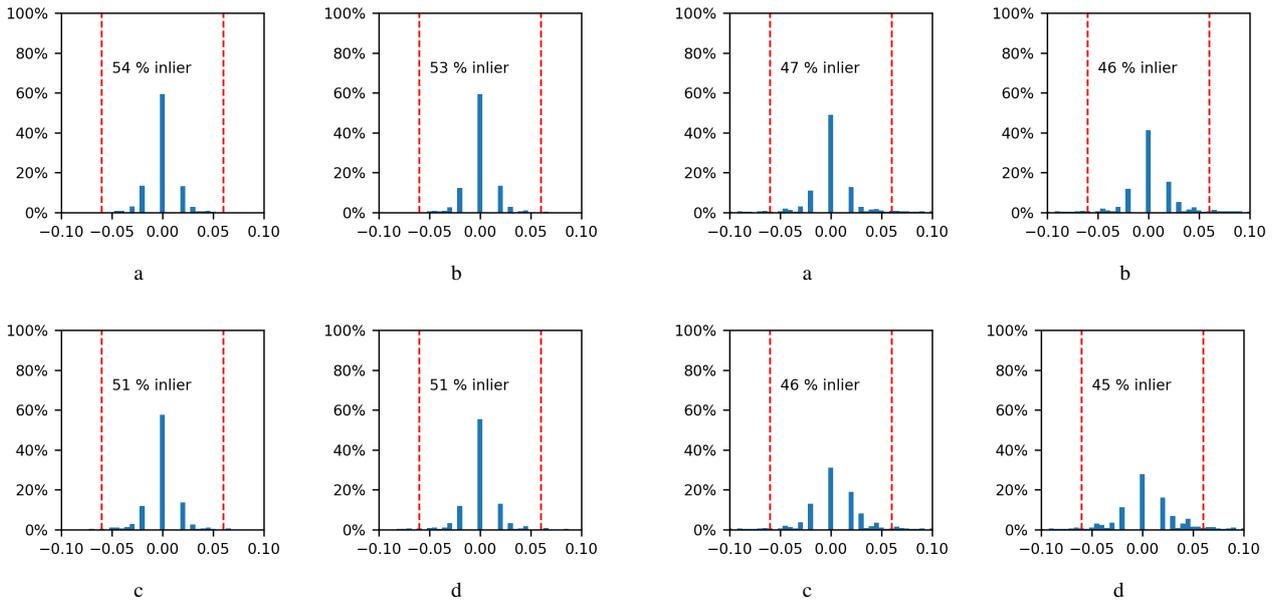


Figure 9. Distance histograms of bounding box *BB 1: Test E9*, *Map E8*, shift-x = 0, (a) shift-y = 0, (b) shift-y = 1 cm, (c) shift-y = 2 cm, (d) shift-y = 3 cm (x-axis: d_v in meters)

Figure 11. Distance histograms of bounding box *BB 2: Test E9* in *Map E8*, shift-y = 0, (a) shift-x = 0, (b) shift-x = 1 cm, (c) shift-x = 2 cm, (d) shift-x = 3 cm (x-axis: d_v in meters)

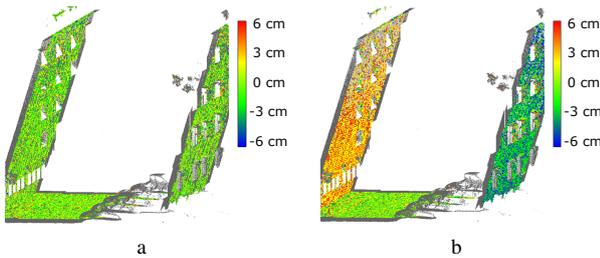


Figure 10. Point cloud of *BB 1, Test E9*: (a) original trajectory, (b) trajectory shifted by 3 cm in x-direction: the shift leads to positive discrepancies (yellow-red colors) on one facade and negative discrepancies (green-blue) on the opposite facade

7. CONCLUSION AND OUTLOOK

We were able to confirm all our hypotheses and have shown that our maps are suitable for precise localization and include seasonal effects. The seasonal maps closest in time to the test set give the best results. We conclude that our dataset is well suited for the creation of high-precision long-term maps and for the future want to improve our mapping method.

As next steps, we want to combine the benefits of the seasonal maps with those of the comprehensive map created from all measurement epochs. To this end we want to cluster the measurement epochs in a meaningful way, so that a seasonal map is composed of several epochs of this season.

In this work we only used a small subset of our data. For the future we want to use the complete temporal extent of our data set. In addition to season effects we will then see dynamics with other periods, e.g. weekly or daily changes, which will occur for different objects. While in this work for the tree in *BB 2* the most suitable map is the one from the same season, for other objects it may be useful to apply a map from the same

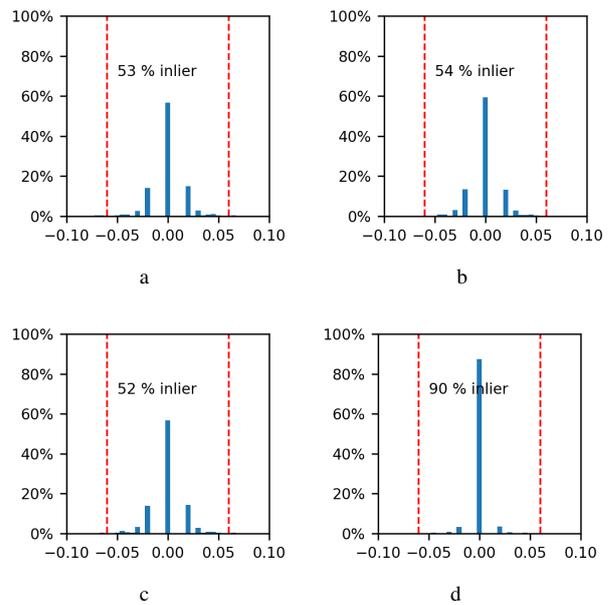


Figure 12. Distance histograms of bounding box *BB 1*, original trajectory, (a) *Map E1*, (b) *Map E8*, (c) *Map E14*, (d) *Map All* (x-axis: d_v in meters)

hour (pedestrians waiting at a bus stop) or day of the week (garbage bins, market stands). In order to find the correct map for every object, we want to combine the temporal maps with a semantic segmentation and learn the temporal behavior of different object classes.

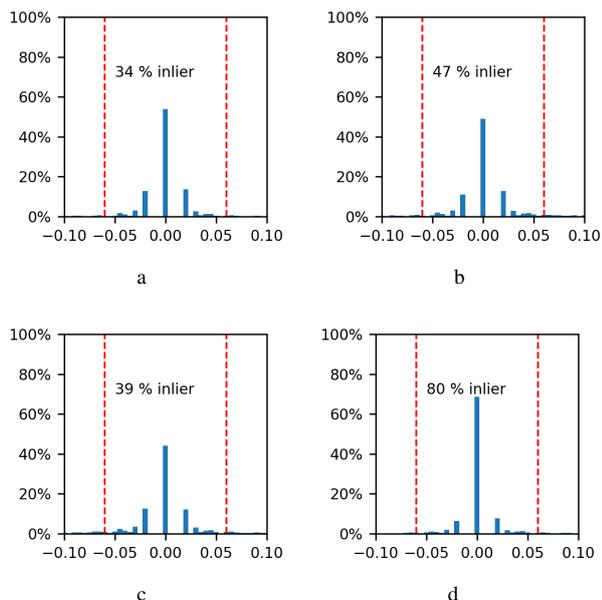


Figure 13. Distance histograms of bounding box *BB 2*, original trajectory, (a) *Map E1*, (b) *Map E8*, (c) *Map E14*, (d) *Map All* (x-axis: d_v in meters)

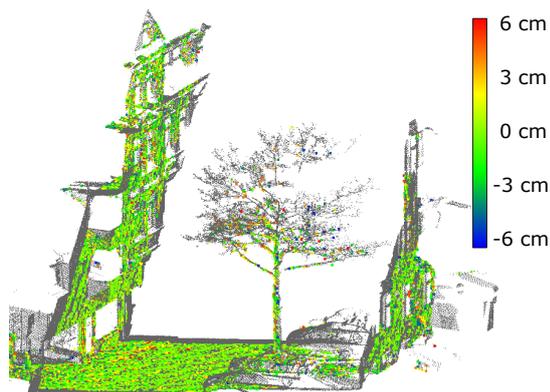


Figure 14. Point cloud of *BB 2*, original trajectory in *Map E1*: low amount of information in treetop in March

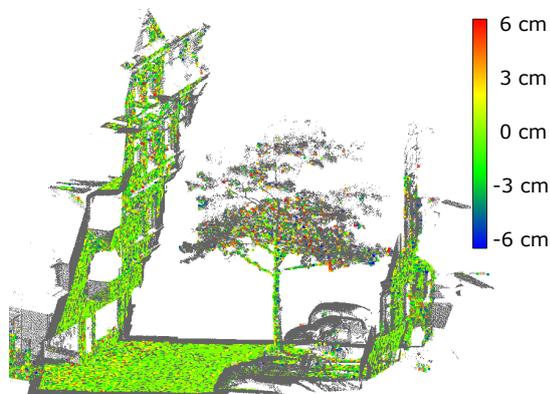


Figure 15. Point cloud of *BB 2*, original trajectory in *Map E8*: comparable amount of vegetation points

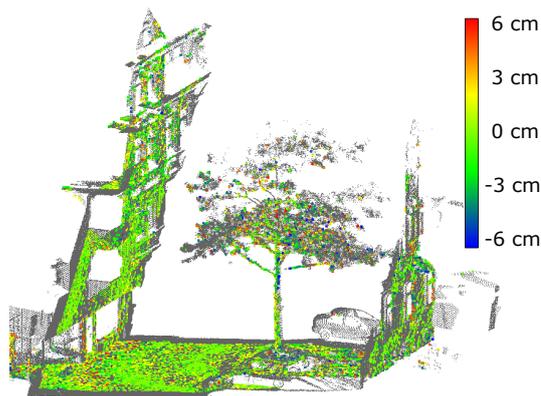


Figure 16. Point cloud of *BB 2*, original trajectory in *Map E14*: comparable or slightly different amount of vegetation points in slightly different places

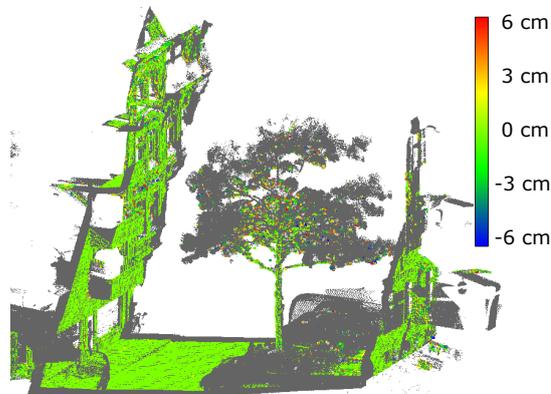


Figure 17. Point cloud of *BB 2*, original trajectory in *Map All*: very high voxel density in the treetop, overlapping vehicles on the parking lane

ACKNOWLEDGEMENTS

This project is supported by the German Research Foundation (DFG), as part of the Research Training Group i.c.sens, GRK 2159, Integrity and Collaboration in Dynamic Sensor Networks.

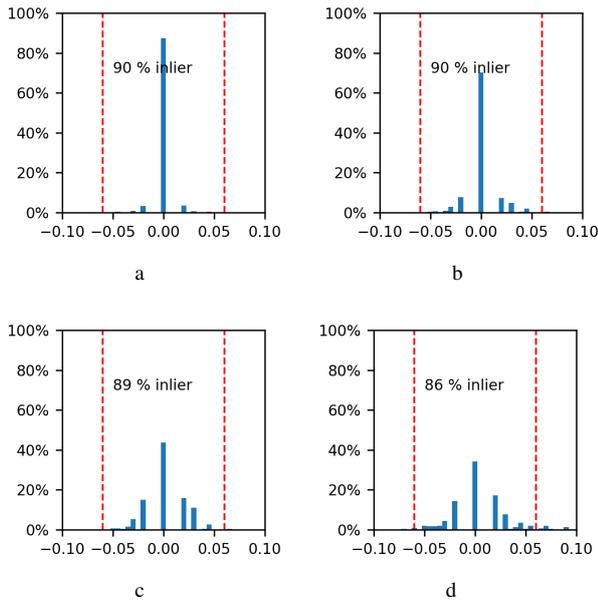


Figure 18. Distance histograms of bounding box *BB 1: Test E9* in *Map All*, shift-*y* = 0, (a) shift-*x* = 0, (b) shift-*x* = 1 cm, (c) shift-*x* = 2 cm, (d) shift-*x* = 3 cm (x-axis: d_v in meters)

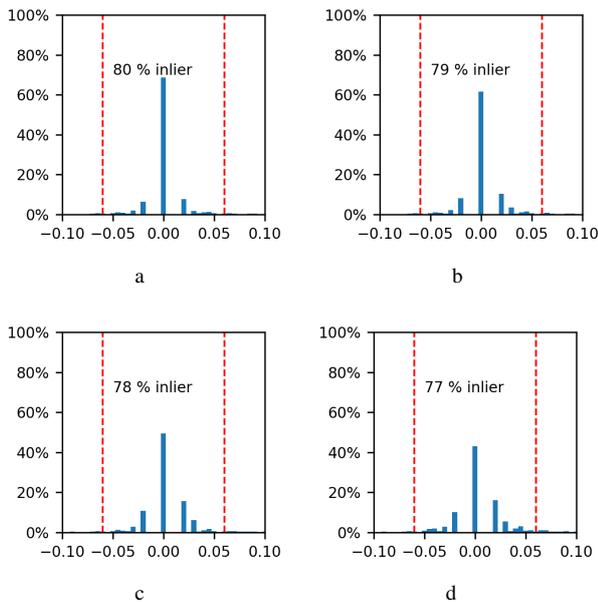


Figure 19. Distance histograms of bounding box *BB 2: Test E9* in *Map All*, shift-*y* = 0, (a) shift-*x* = 0, (b) shift-*x* = 1 cm, (c) shift-*x* = 2 cm, (d) shift-*x* = 3 cm (x-axis: d_v in meters)

References

- Amanatides, J. and Woo, A., 1987. A fast voxel traversal algorithm for ray tracing. In: *In Eurographics 87*, pp. 3–10.
- Biber, P., Fleck, S. and Duckett, T., 2005. 3d modeling of indoor environments for a robotic security guard. In: C. Schmid (ed.), *CVPR*, IEEE Computer Society, Los Alamitos, Calif., p. 124.
- Bock, F., Liu, J. and Sester, M., 2016. Learning on-street park-

ing maps from position information of parked vehicles. In: *Geospatial Data in a Changing World*, Springer, pp. 297–314.

Brenner, C., 2016. Scalable estimation of precision maps in a mapreduce framework. In: M. Ali and S. Newsam (eds), *SIGSPACIAL '16: Proceedings of the 24th ACM SIGSPACIAL International Conference on Advances in Geographic Information Systems*, ACM Press, New York, New York, USA.

Cheng, H. and Sester, M., 2018. Modeling mixed traffic in shared space using lstm with probability density mapping. In: *2018 IEEE Intelligent Transportation Systems Conference*, IEEE, Piscataway, NJ, pp. 3898–3904.

Krajník, T., Santos, J. M. and Duckett, T., 2015. Life-long spatio-temporal exploration of dynamic environments. In: T. Duckett (ed.), *2015 European Conference on Mobile Robots*, IEEE, Piscataway, NJ, pp. 1–8.

Meyer-Delius, D., Beinhofer, M. and Burgard, W., 2012. Occupancy grid models for robot mapping in changing environments.

RIEGL VMX-250, 2012. Technical report, RIEGL Laser Measurement Systems GmbH.

Stachniss, C., 2009. Mapping and localization in non-static environments. In: C. Stachniss, F. Groen, O. Khatib and B. Siciliano (eds), *Robotic Mapping and Exploration*, Springer Tracts in Advanced Robotics, Vol. 55, Springer, Berlin and Heidelberg, pp. 161–175.

Thrun, S., 2002. Robotic mapping: A survey. *Exploring artificial intelligence in the new millennium* 1(1-35), pp. 1.

Thrun, S., Burgard, W. and Fox, D., 2006. *Probabilistic robotics*. Intelligent robotics and autonomous agents series, MIT Press, Cambridge, Massachusetts and London, England.