

High Definition Mapping Using LiDAR Traced Trajectories

STEFFEN BUSCH¹, JANNIK QUEHL² & CLAUS BRENNER¹

Abstract: In this paper, we automatically reconstruct a high definition (HD), lane accurate model through trajectory analysis. The fundamental idea behind our approach is that in the future, many, if not all, of the vehicles will carry sensors, which can be used to keep HD maps up to date. In order to explore this idea, we used a static 3D laser scanner placed at an intersection. In the continuously generated scans, we tracked objects and obtained their trajectories. From these, a map was derived based on trajectory clustering and least squares adjustment. We evaluate the results by comparing them to a ground truth map labeled manually from a mobile mapping LiDAR point cloud.

1 Introduction

Today there are more and more use cases for digital maps. Car navigation systems, land use, vacation trip planning, market analysis and many other web-based services are widespread cases with different requirements with regard to reliability, precision and accuracy. Especially autonomous vehicles, virtual reality (VR) and smart cities will increase the requirement for future maps dramatically. Maps will become more detailed to provide features for automatic localization and behavior recognition. The high dynamics of the environment makes frequent map updates necessary. Features may appear, vanish or change rapidly because of road construction work, accidents, or other unpredictable changes. These changes need to be mapped as soon as they appear to update the virtual representation of the scene. For some applications, like VR or augmented reality for decision making support, there are only moderate requirements with respect to the accuracy and timeliness of the contained information. However, for others, like autonomous vehicles, the map must

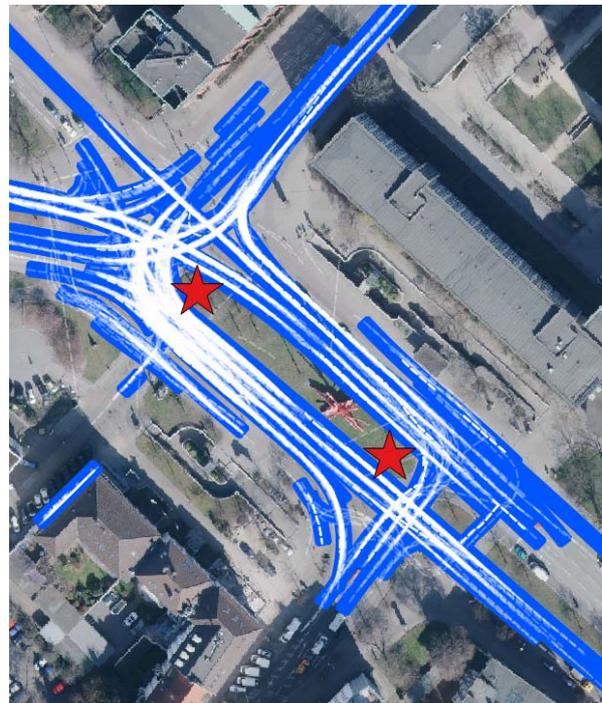


Fig. 1: Data and results of our experiment: trajectories (white), lane segments (blue), scanner positions (red). Ortho image: (HANNOVER 2017)

¹ Leibniz Universität Hannover, Institute of Cartography and Geoinformatics, Appelstraße 9a, D-30167 Hannover, E-Mail: [Steffen.Busch, Claus.Brenner]@ikg.uni-hannover.de

² Karlsruhe Institute of Technology, Institute of Measurement and Control Systems, Engler-Bunte-Ring 21, D-76131 Karlsruhe, E-Mail: Jannik.Quehl@kit.edu

be an accurate, up-to-date representation of the environment, which is constantly compared to the perception of the “smart machines” which use it.

We propose to generate and maintain a representation of the environment in the form of a dynamic high definition map, in a crowd sensing manner through sensor information gathered by the daily traffic. For our analysis, we use trajectories, because highly automated driving will at least provide information about the movement of other traffic participants to assist the drivers. Moreover, trajectories are easily collectable in a crowd-sensing manner from many different sources. In this approach, we focused on a lane accurate map for autonomous vehicles (Fig. 1).

We present our first experiment to derive lanes with an accuracy of a few centimeters by analyzing vehicle trajectories. Since the acquisition of a large number of vehicle trajectories would require us to fit a fleet of vehicles with very accurate positioning sensors, which is beyond our scope, we used a 3D laser scanner instead to observe a specific junction. In more detail, we placed a Velodyne HDL-64 S2 3D laser scanner at a complex junction to track vehicles with a scan frame frequency of 10 Hz. We focused on trajectory analysis because trajectories provide more detailed and dynamic information about the actual traffic behavior than static infrastructure. The resulting trajectories were clustered in order to find the lanes. The lanes were then approximated using a cubic polynomial, fitted by least squares adjustment. We accomplished the geo-referencing by aligning poles, automatically detected in the laser scans, to a reference pole map obtained using a total station. The geo-referenced HD map was evaluated against a manually generated lane model, based on Riegl VMX-250 mobile mapping system measurements.

The paper is structured as follows: In section 2, we give an overview over actual mapping procedures and trajectory based road network construction in detail. In section 3, we will present our approach: firstly, we present our pole geo-referencing procedure. Secondly, we will introduce our clustering and lane alignment procedure. In section 4, we present our dataset. Section 5 evaluates our result by comparing the lane model to a manually generated model. Finally, we will discuss our results and give an outlook in section 6.

2 Related Work

This section gives an overview over the state of the art of mapping approaches. The approaches are divided into *mobile mapping* (MM) and *remote sensing* (RS), which are considered to be associated with a relatively large acquisition cost, versus crowd sensing, which is considered to be a low cost technique. The overview illustrates the need for crowd sensing to achieve high frequent map updates and define the accuracy for high definition maps. The section concludes with the overview of crowd sensed trajectory analysis approaches.

There are plenty of different mapping approaches for different kinds of maps. We focus on the state of the art mapping procedures for commercial navigation systems. Today, most maps are acquired using measurement campaigns, where mobile mapping systems or airplanes gather the required data (TOTH & JÓZKÓW 2016). Thus, the spectrum of data useful for road network reconstruction reaches from aerial imagery, via ego camera images, to high-resolution LiDAR point clouds. On the one hand, RS is more complex but also more efficient for measuring large areas in contrast to the high effort for frequent, area-wide measurements via a fleet of mobile mapping systems. On the other hand, even if UAV holds the key to simplify RS (NEITZEL & KLONOWSKI

2011; TOTH & JÓZKÓW 2016), mobile mapping vehicles generate much more detailed information. MM reaches a resolution down to a few millimeters (PUENTE et al. 2013; GUAN et al. 2016) in contrast to meter or decimeter resolution (TOTH & JÓZKÓW 2016) for airborne measurements. With respect to geo-referencing, all mobile mapping systems depend on their localization accuracy. Geo-referencing is distinguishable into direct (GNSS/IMU based), indirect (control point based) and a combination – the integrated sensor orientation (ISO) approaches. The direct approaches vary between 10 and 0.02 meters accuracy, depending on the sensor quality and correction approach (TOTH & JÓZKÓW, 2016). Although ISO methods are more robust against occlusion, we decided to use full indirect geo-referencing in our first experiment by ensuring the visibility of control points from different static measurement positions. The presented approaches have high data acquisition costs in common and the large amounts of acquired sensor data still need a large amount of manual work in post processing to transform them into maps like the widely used Google or Here maps. An alternative is the use of crowd sensing, discussed in the following.

One of the most famous examples for crowd sensing in the geographic area is Open Street Map (OSM) (M. HAKLAY & WEBER 2008). The members of the OSM community are both users and volunteers. GOODCHILD (2007) characterizes the user responsibility for completeness and quality of geographic information with the term “volunteered geographic information” (VGI). HAKLAY et al. (2013) analyze the commercial use of crowd sourcing and distinguish between passive and active community input. OSM uses the active community input by allowing its community members to change every information, for the price of temporally inconsistent data. In contrast to OSM, commercial providers usually only use data of their users to take care of the data accuracy at every moment. However, widely known providers use their community knowledge to benefit from the self-healing effect of active input. Google produces maps by allowing users to upload their “expert knowledge”, for example, filling gaps or correcting mapping errors. Some commercial providers use a hybrid approach by checking the active community information for consistency before they integrate the knowledge into their maps. Other companies work with passive community information to avoid the complex consistency-check of user expert knowledge. They cherish the users as sensor probes and update their database with traditional tools. We will follow this approach of passive community information by combining sensor data from different independent poses and by analyzing trajectories, which could easily be collected by the crowd (TANG et al. 2016).

Trajectory data generated by crowd sensing have been used in the past to create digital navigation maps (ROETH et al. 2017; RUHHAMMER et al. 2017; DURAN et al. 2016). The analysis methods can be broadly classified into three different approaches: *intersection linking*, *incremental track insertion* and *point clustering* (AHMED et al. 2015). *Intersection linking* first finds a set of intersections by analysis of movement characteristics or point density and afterwards connects these nodes by interpolation. These approaches work best for ‘Manhattan style’ networks. The incremental track insertion tries to match tracks successively to a map, starting with the first track and an empty map. The map becomes more detailed by adding nodes and edges based on the tracks. The point clustering approaches can be distinguished into kernel density and intersection clustering based approaches. Much research was done in the field of network graph generation, whereas relatively few approaches address the generation of highly detailed information,

e. g. about the lane curvature, width or stop line position. Autonomous vehicles and even advanced driver assistance systems rely on highly detailed map information. BENDER et al. (2014) introduce so-called lanelets as map representation for autonomous vehicles. All three approaches, *intersection linking*, *incremental track insertion* and *point clustering*, have the potential to provide high definition maps, if the input data is detailed enough. However, the strict requirements of the navigation data standard (NDS) with respect to accuracy and detail make it necessary to use crowd sensing data; JUNG et al. (2016) use dash cam images for an efficient lane detection. The acquisition of lane-accurate maps from mobile mapping data approaches cannot satisfy the up-to-date requirement for the NDS, *active crowd-sensing* with an error of up to 1.5 m (LANDSIEDEL AND WOLLHERR, 2017) cannot maintain the accuracy to ensure a safe navigation and *hybrid* solutions are too complex and too expensive for area-wide mapping. Several approaches have been proposed to automatically analyze passive crowd-sensing data, like ego-camera images (JUNG et al., 2016; BENDER et al., 2014) and trajectories (ROETH et al., 2017; RUHHAMMER et al., 2017; DURAN et al., 2016) to generate a high definition lane accurate maps. In contrast to ROETH et al. (2017) and RUHHAMMER et al. (2017), JUNG et al. (2016) and RABE et al. (2016) rely on the identification of road markings in sensor data. Nevertheless, the accuracy of trajectory analysis as well as road marking analysis depend to the quality of geo-referencing, even if the alignment methods can improve the accuracy. Our approach uses trajectory data from two different viewpoints, with the advantage that trajectories from one source guarantee a relative precision by LiDAR tracking beyond other geo-referencing tracks (RAZA AND ZHONG 2017; TREIBER & KESTING 2013; MORDECHAI HAKLAY & WEBER 2008; HU et al. 2017; VIVACQUA et al. 2017; KNOOP et al. 2017).

3 Our HD Mapping Approach

Our fully automatic trajectory high definition mapping combines 3D laser scanner measurements from different positions by geo-referencing. Therefore, we track dynamic objects in each scan and map them to their global position in the UTM coordinate frame. After we cluster the tracks by DB-SCAN clustering, we generate lanes by least square alignment; see Fig. 2.

3.1 Geo-referencing

We use a map of poles in the city of Hanover (SCHLICHTING & BRENNER 2014) and a coarse localization to restrict the area for an automatic geo-referencing by an iterative closest point (ICP) algorithm. We determine the position of the poles in the scans by transforming the scan data into a depth image. Therefore, we define a zero degree azimuthal angle (φ) and sort the measurement points into a grid with the resolution of approximately 0.17° for the scanner rotation. The vertical resolution corresponds the 64 vertical rays of the scanner, which is why the resolution of the grid rows is irregular. Finally, we fill in the distance for each measurement into the grid and analyze the resulting depth image. The analysis starts with finding depth jumps row-wise to define azimuthal angles of pole candidates. Next, candidates of different rows are connected by region growing. In the end the candidates are filtered by geometric features; objects wider than 0.5 m and shorter than 2 m are discarded. The resulting poles are matched to the pole map by using the ICP algorithm with various initial setups.

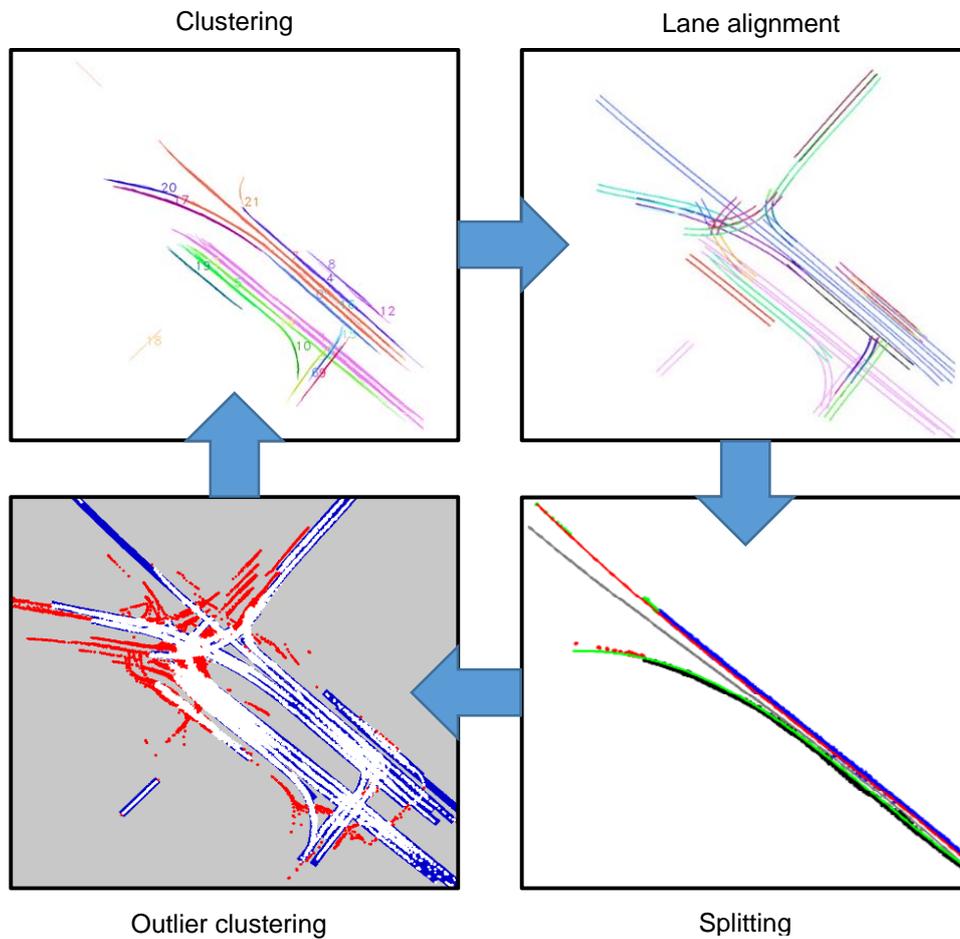


Fig. 2: Main steps of our mapping approach

3.2 Tracking

First, we detected objects in the point clouds from the 3D laser scanner; by removing the ground plane and clustering the remaining points. We estimated the plane by using RANSAC (FISCHLER & BOLLES 1981). The objects were identified by using an Euclidean cluster extraction provided by the point cloud library (RUSU & COUSINS, 2011). The cluster centers of gravity of the convex hulls determined the position of the object. We generated the trajectories (see Fig. 3) by using a *Unscented Kalman Filter* with a CYRA (HOUEYOU et al. 2013; SCHUBERT et al. 2008) motion model and associated objects of successive time steps with the nearest neighbor

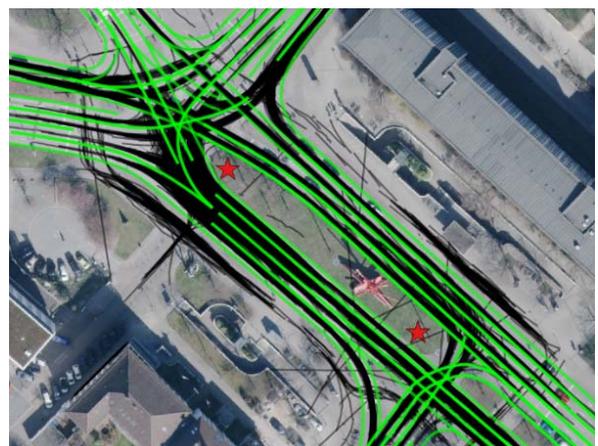


Fig. 3: Trajectories (black) and reference lane markings (green). Ortho image: (HANNOVER 2017)

approach.

3.3 Lane Clustering

We assume that each track represents part of a lane and can be approximated by a Hermite Cubic Spline (*HCS*) (CATMULL & ROM, 1974). This assumption allows for an efficient distance metric and clustering. Our approach uses two successive clustering steps for road segments that differ a lot in density and frequency, e.g. straight lanes and curves. First, we cluster the clear, less curvy and frequently used road segments with a cluster distance of 0.3 m. Secondly, we cluster the outlier trajectories, which do not fit the previously generated lane model with a cluster distance of 0.5 m (see Fig. 2 **Fehler! Verweisquelle konnte nicht gefunden werden.**, *outlier clustering*). We calculate the cluster distance for the DB-SCAN clustering by transforming each trajectory from the world frame into a *HCS* frame by translating the origin into the middle of the trajectory and rotate the axis by the angle of a vector v . This vector v is calculated by the difference between the end and start point of the trajectory. This transformation is used to transform the start and end points as well as their gradient of trajectories to determine the *HCS*s. The cluster distance is calculated from 5 points from one *HCS* by transforming the world coordinates into the frame of the other *HCS* replacing the y coordinate by its function value and transforming the point back into the world to calculate the Euclidean distance. In case one point is outside the definition area of the *HCS* we extrapolate in a linear manner to calculate the new y value.

3.4 Lane Alignment

The lane segment geometry is estimated using a cubic polynomial by a least squares adjustment. First, the start and end point for the trajectory cluster are determined by analyzing the cluster direction via principal component analysis of all trajectory points. Secondly, the temporary direction vector is used to define the first start and last end point. Finally, these start and end points define vector v (1), which determines the trajectory cluster center o (2).

$$v_i = e_i - s_i \quad (1)$$

$$o = s_i + \frac{v_i}{2} \quad (2)$$

With v , u (3) and o each trajectory point of one cluster is transformed into the local (“polynomial”) frame by (4).

$$u_i = \begin{pmatrix} -v_{2,i} \\ v_{1,i} \end{pmatrix} \quad (3)$$

$$x' = \frac{v^T}{\|v^T\|} (p - o)$$

$$y' = \frac{u^T}{\|u^T\|} (p - o) \quad (4)$$

The least squares adjustment (5) estimates the lane polynomial for a cluster.

$$\hat{s} = (A^T A)^{-1} A^T l \quad (5)$$

With:

$$\hat{s} = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}, A = \begin{pmatrix} x_1^3 & x_1^2 & x_1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_i^3 & x_i^2 & x_i & 1 \end{pmatrix} \text{ and } l = \begin{pmatrix} y_1 \\ \vdots \\ y_i \end{pmatrix}$$

Clustering errors are detected by filtering each aligned lane segment with an RMSE () above 0.5 m.

$$rmse = \sqrt{(Ax - l)^T (Ax - l)} / i \quad (6)$$

The clustered trajectory points of incorrect lane segments are split into points below and above or points left and right. Thus, four lanes are aligned by allocating each point to two of the four sets left, right, above and below if the x value belongs to the first or second part or the y value is above or below the function value. The pair (left/right or above/below) with the lower RMSE is replacing the splitting segment. Fig. 2, *splitting*, illustrates the splitting step for one of the clusters. This procedure is repeated until no more cluster error is detected. Our splitting procedure can generate some duplicates of the same lane segments in some cases, for example, if more than one split is required to distinguish neighboring lanes. Thus, four segments describe two real world lanes. All generated duplicates are identified by an overlap of over 50% and direction difference below 20° and get merged by estimating the merged lane through the union of the pseudo lane trajectory points. Next, we assume a general lane width of 3 m and look for outlier trajectories from our preliminary road lane model. Therefore, we split each trajectory and the entrance and leaving points of our lanes and filter the trajectory snippets outside our model, see Fig. 4, *outlier clustering*.

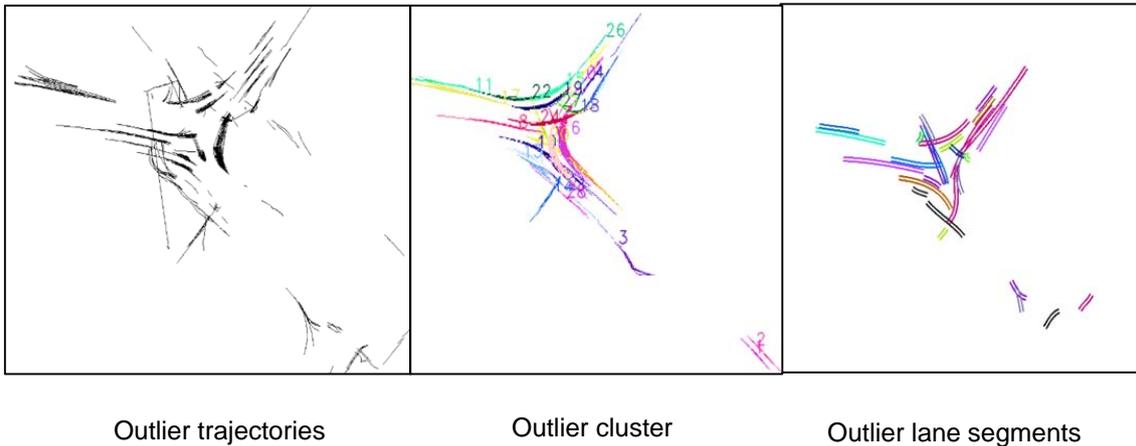


Fig. 4: Outlier trajectory clustering

Fig. 4 shows the outlier trajectory clustering with less strictly set cluster parameters, cluster distance of 0.5 m in contrast to 0.3 m and the corresponding aligned lane segments. Finally, our lane accurate HD-Map is generated by merging both sets of lane segments and removing the overlapping segments.

4 Experiment

We used a total station, a mobile mapping system (Riegl VMX-250), and a 3D laser scanner (Velodyne HDL-64 S2) to map a complex junction, *Königsworther Platz* in Hannover. First, we measured poles around the area of the *Königsworther Platz* to generate a map of poles for the geo referencing procedure (BRENNER & HOFMANN, 2012). Secondly, we did several mobile mapping measurements from different directions along the junction. Finally, we recorded the traffic at the junction for two hours with the 3D laser scanner; see Fig. 5. We used two different positions for the measurements to generate trajectories that are more comprehensive and in order to evaluate our automatic geo-referencing approach. Thus, we generated 72000 (36000 per scan pose) point clouds and 4621 trajectories with a length of more than 10 m.

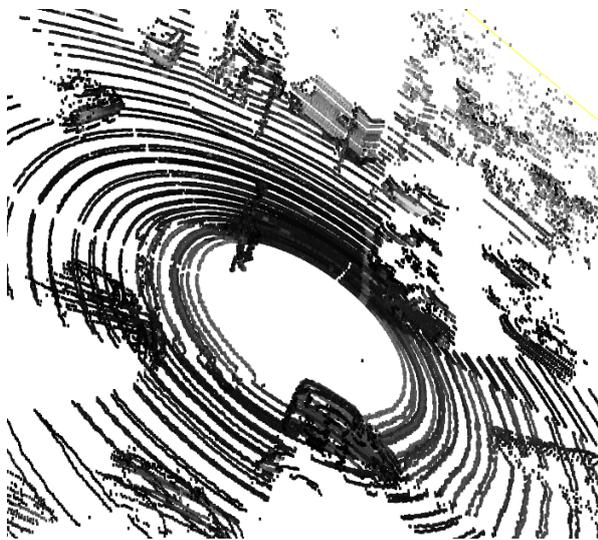


Fig. 5: Velodyne data

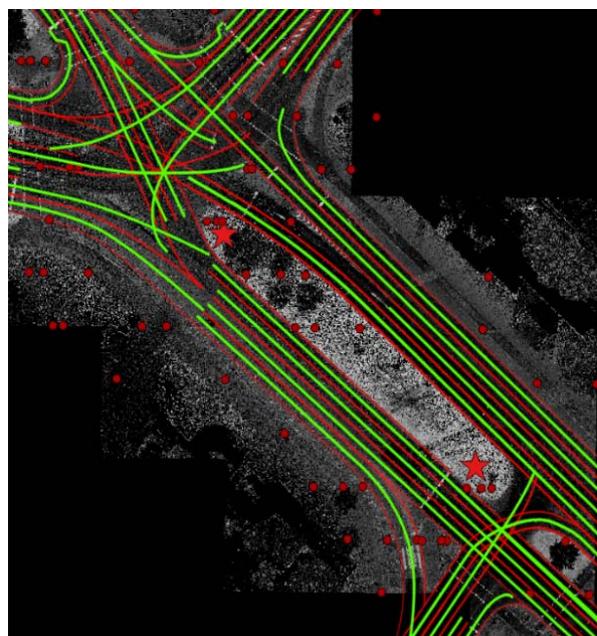


Fig. 6: Reference data lane borders (red lines), middle axis (green lines) and poles (red points)

4.1 Static reference data

Fig. 6 shows the 91 poles (red dots) which were measured by a total station with an accuracy better than 10 cm (BRENNER & HOFMANN 2012). Several mobile measurement drives were used to generate a highly detailed point cloud with 165.6 million points. We picked 1956 points from this point cloud at lane marking and curbs to estimate 35 lane segments with their borderlines (red) and middle axes (green). In the two cases where one lane border was not clearly defined by markings or curbs, we interpolated the lane border with a width of 3 m. Afterwards, we calculated the middle axis of each lane segment to evaluate our automatically detected middle axis.

5 Evaluation

First, we evaluated the geo-referencing of the scanner positions by calculating the root mean square error between the automatically detected poles and the manually measured poles. Secondly, we calculated the error between the automatically detected and the manually determined lane middle axis.

5.1 Geo-referencing

Fig. 7 shows the scanner positions (red stars) which were generated by calculating the transformation of the 21 automatically detected pole positions (blue circles) to the 91 manually measured pole positions (white circles). The RMSE errors for positions are 0.05 m and 0.07 m.

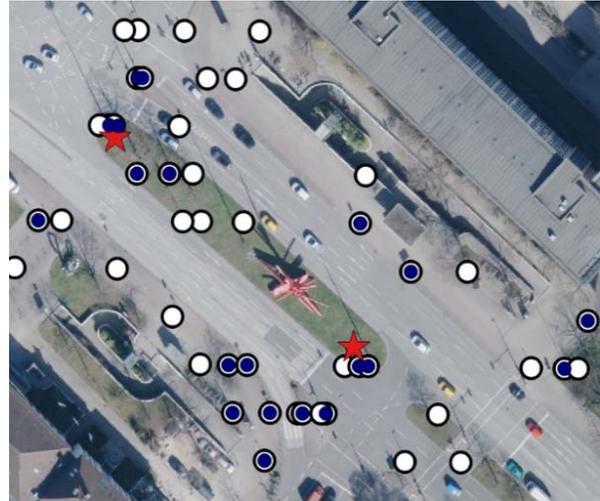


Fig. 7: Geo-referencing, manually measured poles (white circle), automatically detected poles (blue circle) and scanner position (red stars). Ortho image: (HANNOVER 2017)

5.2 Map

We evaluated the quality of our lane accurate map by comparing the lane middle axis based on the manually detected points (mp) at lane markings and curbs to the automatically detected middle axis grid in 10 cm steps. Fig. 8 shows the histogram of the distance for each manually detected point to the nearest automatically detected lane (left) and the distance for the automatically detected middle axis point to the nearest manually detected middle axis lane (right). The large distances reflect the challenge of defining the center of vehicles tracked by a laser scanner. Depending on scanning a vehicle frontally or from the side, the center varies up to half the vehicle width. Fig. 9 shows that the biggest error occurs at the border of the evaluation or tracking area and at areas with missing tracks and additional tracks from pedestrians and cyclists. Moreover, the direct alignment of the

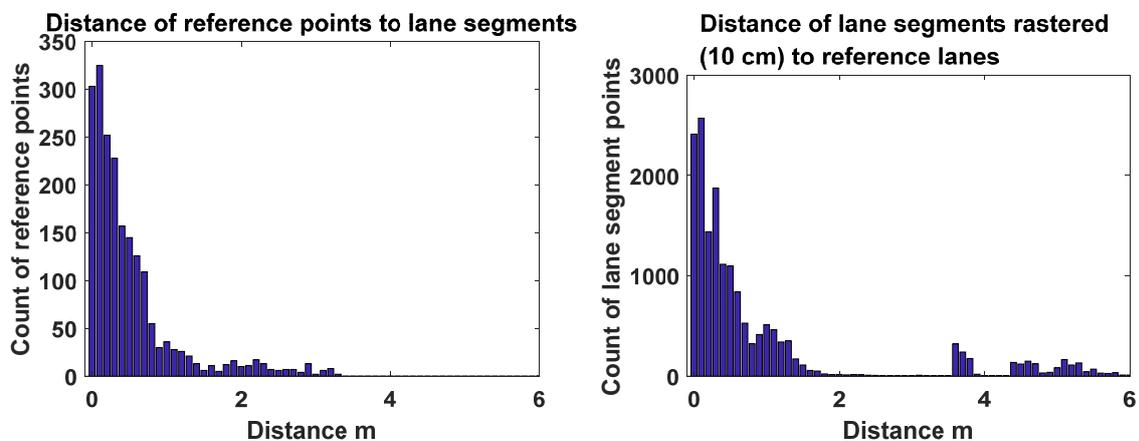


Fig. 8: Histograms of distance between reference lanes and automated lanes distance for each reference point (left) and distances for the automated lane segments (right).

middle axis based on trajectories could not compensate for common driving maneuvers illustrated in the center of the junction; during a turn, vehicles do not drive in the middle of the lane. The histogram (Fig. 8) shows that more than 500 m middle axis were automatically detected in the global UTM coordinate frame with less than 20 cm error.

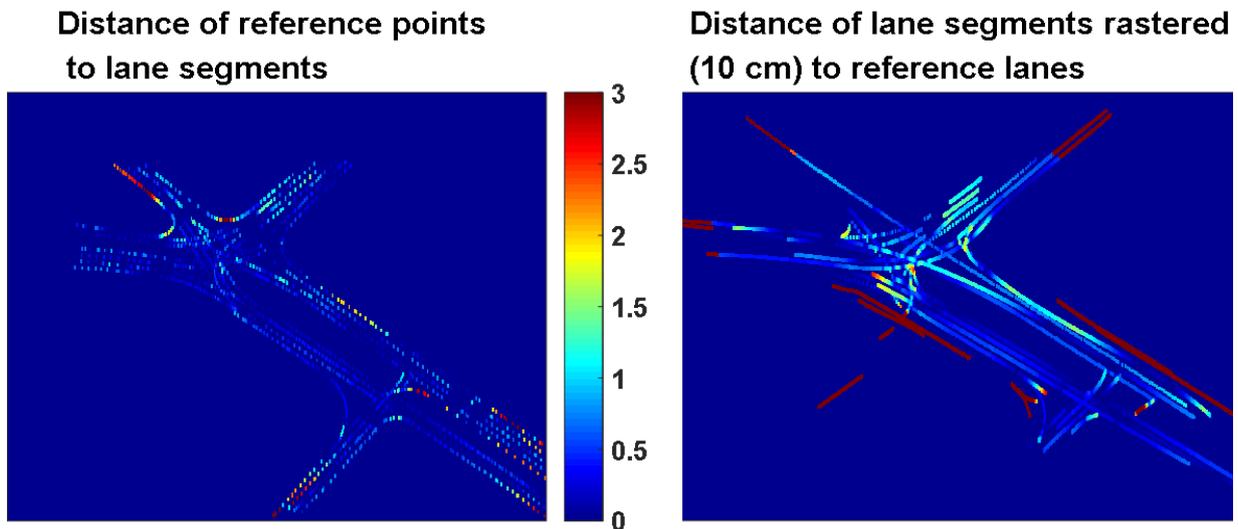


Fig. 9: Heat map of error for reference points (left) and lane segments (right), clamped in the interval 0 to 3 m

6 Discussion and Outlook

We discussed the quality of the result, the problems with mapping the middle axis of lanes by 3D laser scanner measurements and detection problems. We end with an outlook of the problems we addressed and the way we plan to improve our HD-Mapping by analyzing trajectory data.

6.1 Discussion

The problem of missing tracks could be solved by more reliable detections and a more comprehensive measurement via crowd sensing. The mix of vehicles, pedestrians and cyclists participating in the traffic could be distinguished by a classification of detections or other filter solutions. The exact middle positions of detected vehicles could be determined by aligning all point clouds for each detected track and estimating an active shape model to this data (COENEN et al. 2017). Middle axis differences because of driving behaviors could be addressed by generating a more model-based view of a junction. The high accuracy of below 20 cm for 34% percent of our aligned lane segments, ignoring wrong assignments (error > 3 m), shows the potential to generate lane accurate models from trajectories of crowd sensed data like laser scanner measurements.

6.2 Outlook

We plan to complete missing tracks and distinguish traffic participants by improving our detection procedure with pixel segmentation at the depth images by a neuronal network. Thus, we will improve the quantity and accuracy of the trajectories. Furthermore, we will solve the problem of

varying driver behavior and uncertainties by estimating more general junction models using a Markov Chain Monte Carlo optimization. Finally, we will use several vehicles equipped with Velodyne laser scanners to evaluate our approaches in larger inner city areas.

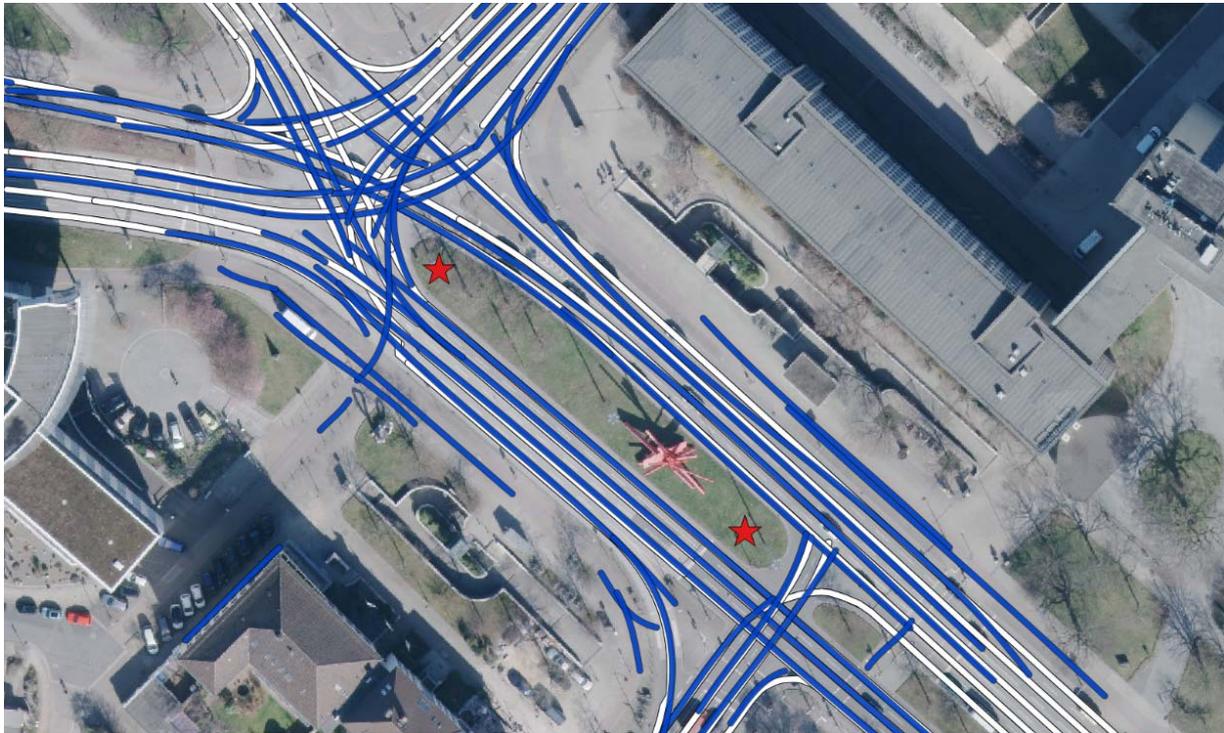


Fig. 10: Comparison of manually labeled (white) and automatic detected (blue) lane axes. Ortho image: (HANNOVER 2017)

7 ACKNOWLEDGEMENTS

Steffen Busch and Jannik Quehl were funded by the German Science Foundation DFG within the priority program SPP1835, “Cooperative Interacting Automobiles”.

8 References

- AHMED, M., KARAGIORGOU, S., PFOSE, D. & WENK, C., 2015: A comparison and evaluation of map construction algorithms using vehicle tracking data. *GeoInformatica* 19, 601-632.
- BENDER, P., ZIEGLER, J. & STILLER, C., 2014. Lanelets: Efficient map representation for autonomous driving. in: *Intelligent Vehicles Symposium Proceedings*, IEEE, 420-425.
- BRENNER, C. & HOFMANN, S., 2012: Evaluation of Automatically Extracted Landmarks for Future Driver Assistance Systems. in: Yeh, A.G.O., Shi, W., Leung, Y., Zhou, C. (Eds.), *Advances in Spatial Data Handling and GIS*. Springer Berlin Heidelberg, 169-181.
- CATMULL, E. & ROM, R., 1974: A class of local interpolating splines. in: *Computer Aided Geometric Design*. Elsevier, 317-326.

- COENEN, M., ROTTENSTEINER, F. & HEIPKE, C., 2017. Detection and 3D modelling of vehicles from terrestrial stereo image pairs. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* XLII-1/W1, 505-512. <https://doi.org/10.5194/isprs-archives-XLII-1-W1-505-2017>
- FISCHLER, M.A. & BOLLES, R.C., 1981: Random sample consensus a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **24**, 381-395.
- GOODCHILD, M.F., 2007. Citizens as sensors: the world of volunteered geography. *GeoJournal*, **69**, 211-221.
- GUAN, H., LI, J., CAO, S. & YU, Y., 2016. Use of mobile LiDAR in road information inventory: a review. *Int. J. Image Data Fusion*, **7**, 219-242. <https://doi.org/10.1080/19479832.2016.1188860>
- HAKLAY, M., 2013. Citizen Science and Volunteered Geographic Information: Overview and Typology of Participation. in: *Crowdsourcing Geographic Knowledge*. Springer, Dordrecht, 105-122. https://doi.org/10.1007/978-94-007-4587-2_7
- HAKLAY, M. & WEBER, P., 2008. OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Comput.*, **7**, 12-18. <https://doi.org/10.1109/MPRV.2008.80>
- HANNOVER, 2017: Orthophoto Hannover Koenigsworther Platz [WWW Document]. URL <https://www.hannover.de/Leben-in-der-Region-Hannover/Verwaltungen-Kommunen/Die-Verwaltung-der-Landeshauptstadt-Hannover/Dezernate-und-Fachbereiche-der-LHH/Baudezernat/Fachbereich-Planen-und-Stadtentwicklung/Geoinformation/Open-GeoData/Digitale-Orthophotos-DOP20> (accessed 1.26.18).
- HOUENOU, A., BONNIFAIT, P., CHERFAOUI, V. & YAO, W., 2013: Vehicle trajectory prediction based on motion model and maneuver recognition. in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 4363-4369.
- HU, M., ZHONG, Z., NIU, Y. & NI, M., 2017: Duration-Variable Participant Recruitment for Urban Crowdsourcing With Indeterministic Trajectories. *IEEE Trans. Veh. Technol.* **66**, 10271-10282. <https://doi.org/10.1109/TVT.2017.2718043>
- JUNG, S., YOUN, J. & SULL, S., 2016: Efficient lane detection based on spatiotemporal images. *IEEE Trans. Intell. Transp. Syst.*, **17**, 289-295.
- KNOOP, V.L., DE BAKKER, P.F., TIBERIUS, C.C. & VAN AREM, B., 2017: Lane Determination With GPS Precise Point Positioning. *IEEE Trans. Intell. Transp. Syst.*, **18**, 2503-2513
- LANDSIEDEL, C. & WOLLHERR, D., 2017: Road geometry estimation for urban semantic maps using open data. *Adv. Robot.*, **31**, 282-290. <https://doi.org/10.1080/01691864.2016.1250675>
- NEITZEL, F. & KLONOWSKI, J., 2011: Mobile 3D mapping with a low-cost UAV system. *Int Arch Photogramm Remote Sens Spat Inf Sci.*, **38**, 1-6.
- PUENTE, I., GONZÁLEZ-JORGE, H., MARTÍNEZ-SÁNCHEZ, J. & ARIAS, P., 2013: Review of mobile mapping and surveying technologies. *Measurement*, **46**, 2127-2145.
- RABE, J., NECKER, M. & STILLER, C., 2016: Ego-lane estimation for lane-level navigation in urban scenarios. in: *Intelligent Vehicles Symposium (IV)*, IEEE, 896-901.

- RAZA, A. & ZHONG, M., 2017: Hybrid lane-based short-term urban traffic speed forecasting: A genetic approach. 2017 4th International Conference on Transportation Information and Safety (ICTIS). 271-279. <https://doi.org/10.1109/ICTIS.2017.8047776>
- ROETH, O., ZAUM, D. & BRENNER, C., 2017: Extracting Lane Geometry and Topology Information from Vehicle Fleet Trajectories in Complex Urban Scenarios Using a Reversible Jump Mcmc Method. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **4**, 51.
- RUHHAMMER, C., BAUMANN, M., PROTSCHKY, V., KLOEDEN, H., KLANNER, F. & STILLER, C., 2017: Automated Intersection Mapping From Crowd Trajectory Data. *IEEE Trans. Intell. Transp. Syst.* **18**, 666-677.
- RUSU, R.B. & COUSINS, S., 2011: 3D is here Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation (ICRA)*, 1-4.
- SCHLICHTING, A. & BRENNER, C., 2014: Localization using automotive laser scanners and local pattern matching, in: *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 414-419.
- SCHUBERT, R., RICHTER, E. & WANIELIK, G., 2008. Comparison and evaluation of advanced motion models for vehicle tracking. *11th IEEE International Conference on Information Fusion*, 1-6.
- TANG, L., YANG, X., DONG, Z. & LI, Q., 2016: CLRIC: collecting lane-based road information via crowdsourcing. *IEEE Trans. Intell. Transp. Syst.*, **17**, 2552-2562.
- TOTH, C. & JÓZKÓV, G., 2016: Remote sensing platforms and sensors: A survey. *ISPRS J. Photogramm. Remote Sens.*, Theme issue "State-of-the-art in photogrammetry, remote sensing and spatial information science", **115**, 22-36. <https://doi.org/10.1016/j.isprsjprs.2015.10.004>
- TREIBER, M. & KESTING, A., 2013: Trajectory and Floating-Car Data, in: *Traffic Flow Dynamics: Data, Models and Simulation*. Springer Berlin Heidelberg, Berlin, Heidelberg, 7-12. https://doi.org/10.1007/978-3-642-32460-4_2
- VIVACQUA, R.P.D., BERTOZZI, M., CERRI, P., MARTINS, F.N. & VASSALLO, R.F., 2017: Self-Localization Based on Visual Lane Marking Maps: An Accurate Low-Cost Approach for Autonomous Driving. *IEEE Trans. Intell. Transp. Syst.*, 1-16. <https://doi.org/10.1109/TITS.2017.2752461>