# Automatic Lane-level Road Network Graph-generation from Floating Car Data

Mario Dolancic

University of Salzburg, Austria

## Abstract

While common digital road network graphs are able to represent real-world street network topology relations quite adequately, they are highly generalized with regard to the composition of a road. Irrespective of their actual number of lanes, roads are shown as just one single line. As many intelligent transportation systems (ITS) applications require or provide lane-specific data and services, this is no longer sufficient from a short- to medium-term perspective. In particular, automated driving requires high-accuracy graphs both in topology and in geometry to localize positions not only on the correct road, but also in the correct lane. In the following paper, a cost-effective methodology for deriving such lane-level road network graphs will be described. The methodology is applied to standard GNSS trajectories collected for three different road types (urban, interurban, motorway) by vehicles participating in real-world traffic situations (Floating Car Data). The methodology extracts the number and position of lane centrelines from pre-processed GNSS trajectories using a kernel density estimation (KDE) and distance relations. Results show that the proposed method can, depending on the quality of the input data, reliably model lane centrelines for different road settings.

## Keywords:

high accuracy street maps; Automated driving; GNSS-data

## 1    Introduction and Motivation

Today automotive navigation systems are widely used, both by professionals in the transportation industry and by private consumers. Many vehicle manufacturers offer a GNSS navigation device as a vehicle equipment option, making it an integral part of the vehicle. Moreover, supplementary equipment such as portable navigation devices (PNDs) and smartphones with navigation apps installed are very common. Important components of such navigation units (besides the GNSS receivers themselves) are the quality and accuracy of the underlying street network maps. The common navigation map basically represents the street network as a directed vector graph and is stored in different data formats. A digital graph consists of a set of nodes (or, depending on the terms used, vertices or points) and a set of edges (or arcs or lines), where each edge links two nodes. As a consequence, a road network can naturally be represented by graphs, in which nodes are equivalent to

intersections, addresses, road junctions and dead-ends, while edges represent corresponding roads linking such points. In the real world, edge directions are equivalent to one-way or two-way roads. By further attributing information about length and speed limit to the edges, the digital representation of a street network is enhanced (Thomson & Richardson, 1995).

However, while today's road maps for developed countries have reached good to excellent representation of the roads as a network , the geometric depiction of the roads is highly generalized. Typically, a single line represents a whole road segment without showing the separate lanes. The process of reducing the complexity of physical real-world objects while at the same time trying to preserve the semantic and structural features and the logical relations of these objects is called generalization. A good generalization aims to preserve as much accuracy as possible for the required purpose (Weibel & Jones, 1998). In the context of vehicle navigation, the main requirement is to locate the position of the vehicle on the right road segment. Current road network graphs (Figure 1) seem just sufficient for this task.

However, demand changes over time. The transportation sector and political agendas promote and foster intelligent transportation systems (ITSs) more and more. ITSs describe technology applied to vehicles and infrastructure in order to transfer information between systems for the improvement of safety, transportation performance and travel comfort, and to decrease negative environmental impact. This includes vehicle-to-vehicle and vehicle-to-infrastructure communication as well as in-vehicle applications, in particular Advanced Driver Assistance Systems (ADAS) installed in the vehicle console (Faouzi et al., 2011). In this context, so-called enhanced, content-rich (e.g. lane number, street curvature) maps with improved geometry come into play. Enhanced maps do not simply improve navigation systems by providing more information; they go beyond that. They have proved beneficial to many ITS and especially ADAS applications such as adaptive cruise control, advanced traffic information, speed limit assistance, lane-keeping assistance, curve speed warning, route guidance and path prediction (Pandazis, 2002). In this broader perspective, different national and international research efforts like ED Map (EDMap Consortium, 2004) in the US and the EU's NextMAP (Pandazis, 2002) project particularly emphasize a short-to-medium term necessity for lane-level digital road maps, map-matching technology and information. Examples for such use cases would be lane pre-selection (straight, left/right turn), lane-specific access restrictions (e.g. bus lane), faster and easier formation of an emergency lane, lane-level road conditions (e.g. snow cleared or not), and obstacle warnings (e.g. stranded vehicles or the end of a tailback). Furthermore, such enhanced road maps can improve automated driving by extending the vehicles "range of vision", providing the opportunity literally to look around a corner/bend before other on-board sensors are able to get even a glimpse of the situation. The map would therefore serve as an additional a priori sensor (Lu et al., 2005; Gomes, 2014).

All parties involved – the major players of street map providers HERE®[1], TomTom®[2] and Google®[3], the automotive industry, as well as politicians – have recognized the need for

---

[1]https://company.here.com/automotive/intelligent-car/here-hd-live-map/ [accessed 2016-01-21]
[2]http://automotive.tomtom.com/en/highly-automated-driving/highly-detailed-map [accessed 2016-01-21]
[3]https://googleblog.blogspot.co.at/2010/10/what-were-driving-at.html [accessed 2016-01-21]

high-resolution road graphs (EDMap Consortium, 2004). The first two parties conduct vast measurement campaigns using their vehicle fleets equipped with high-accuracy measurement technology (e.g. GNSS, LiDAR, Image Processing).

However, conducting such measurement campaigns and post-processing the data is expensive and time-consuming. Calculations based on the experience acquired in the EDMap project estimate the time needed to map all road types in the US for lane-level applications to be roughly 10 years. If one assumes a measurement fleet of 20 cars, with each car driving 200 miles per day, and 240 working days per year, to map the existing 5,000,000 road miles in the US, the total miles to be driven would come to 11,000,000. In this calculation, the estimate of 10 years does not even include time for data post-processing or fleet maintenance. Although the data-collection time can be reduced greatly by increasing the size of the fleet, costs of about $150,000 just for the additional measuring equipment are surely a limiting factor (Pandazis, 2002).

Even though these numbers have to be handled with caution, the question arises of which user groups will have access to such kinds of maps besides the providers themselves and the automotive industry (HERE® now belongs to a consortium of Audi, BMW and Daimler) and at what price. Furthermore, data collection in rural areas or whole countries could be postponed or even skipped if economic viability is taken into consideration.

Hence, an affordable solution is necessary to derive accurate lane-level road network graphs (see Figure 1) mainly from standard GNSS-trajectories (e.g. smartphones) of regular traffic participants (Floating Car Data = FCD). In this paper, a data-driven approach which uses statistical methodologies for achieving the goal will be presented. But first, a brief overview of the current state of research and the underlying data for the research will be given.
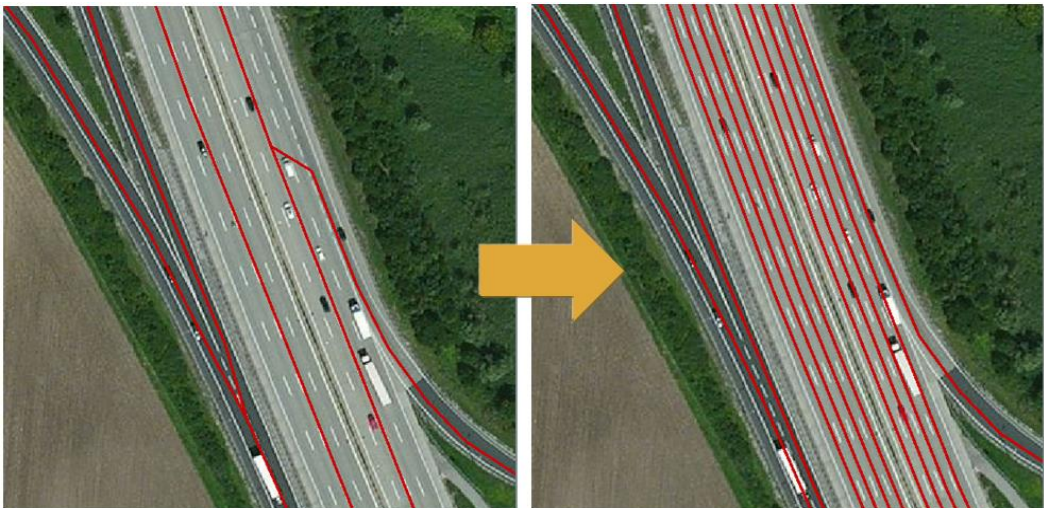


**Figure 1:** Comparison of a conventional road network graph showing single-road geometry (left) and an enhanced road network graph which models each lane separately as separate geometry (right). Background map: Bing Aerial Imagery

# 2    State of research

Extracting topology, geometry and other information from standard GNSS trajectories for the creation and/or refinement of digital street maps has been the topic of several research projects. These projects vary in terms of data and methodology used, as well as their objectives.

Zhang et al. (2010) implemented a cluster algorithm for centre-line generation from collected GNSS traces available over OpenStreetMap (OSM). After some data pre-processing which eliminates the most striking GPS deviations from the data set, the road centrelines are derived from the remaining tracks. Although the number of lanes was estimated, their precise position was not determined, nor was their geometry modelled. Another methodology, which uses a modified Gaussian mixture model (GMM) and was developed by Chen & Krumm (2010), could under the assumption of constant lane widths reliably identify the number of lanes. However, they, too, did not try to localize the lane positions. Edelkamp & Schrödl (2003) tried just that by using a multi-step approach. They start by filtering the input data for GPS noise and outliers. Second, the remaining trajectories are divided into sequences of segments which relate to an initial digital road map, using a modified best-fit path-search algorithm. Third, using a spline-fitting algorithm, they calculate a road centreline for each segment as a reference line for finding other lanes. Last but not least, they determine the number and position of lanes by using a hierarchical agglomerative clustering algorithm. Depending on the quality of the input data (GPS noise), this approach yields errors ranging from less than 5% to over 35% in the number of lanes detected. For their research project, high-precision differential GPS data was used, resulting in a better overall distribution and accuracy of the GPS trajectories. Uduwaragoda at al. (2013) used a probability density function for deriving lane geometries from GPS data. To do this, the Kernel Density Estimations (KDEs) of the GPS tracks are analysed at their intersection points with the perpendiculars and the positions with the highest density values were marked as lane centrelines. Since no reference data was available for error assessment, a visual interpretation of the quality was conducted, which showed good algorithm performance with a minimum input of 150 GNSS tracks.

# 3    Methodology

In most cases, those studies which aimed to extract both the numbers of lanes from GNSS tracks and their precise location used either simulated or enhanced input data with higher positional accuracy than standard GNSS receivers. Furthermore, the evaluation was often only conducted for one use case, most often for motorways. However, the methodology presented in this paper will be applied to the three different road-types: motorways, interurban roads and urban roads.

## Study areas and data set

The input data for this project was collected in three different study areas (Figure 2) near the city of Graz, Austria, over a period of five months in 2015. The study areas include:

- A 3.4 km-long section of urban road with two lanes
- An interurban road, about 12 km in length, with 2 lanes
- A7 km section of motorway with 3 lanes, and a short 2-lane section

The measurements in each study area were collected in both driving directions and can be thought of as Floating Car Data (FCD). FCD is collected by vehicles participating in real-world traffic situations and can – in addition to positional and time information – also include other vehicle information (e.g. steering angle) when connected directly to the vehicle electronics (Krampe, 2007). FCD can be collected by various devices ranging from inexpensive common GNSS-loggers and smartphones to costly high-accuracy DGPS devices.

For this project, different measuring devices, which varied in their spatial and temporal resolution, were used to obtain input GNSS trajectories for the algorithm. Among them was a high-precision GPS-ADMA (Automotive Dynamic Motion Analyser) system with a 100Hz measuring frequency and an average accuracy of <3 cm. Two systems from Racelogic® were also used: one with a 10Hz recording interval and a horizontal accuracy of about three meters; the second system with a 100Hz recording interval and a positional accuracy of approximately two metres. In addition to these, a GNSS-Logger from Qstarz® was used for measurements. With this logger, GNSS tracks were recorded partly with a measuring frequency of 5Hz and partly with a 1Hz measuring frequency, and an accuracy of about three meters. Additionally, a number of GNSS-logger applications were used on various common smartphones, including Sony Xperia Z3/S, Samsung Galaxy S2 and iPhones. Regarding the smartphone data, only the GNSS-Logger app that was used is known, but not the built-in GNSS chip. The GNSS quality of the different smartphones and apps varied regarding spatial accuracy, but was situated mostly within the range of <5m, with a measuring rate of 1Hz over all devices.

For better GNSS reception, all GNSS systems were placed on or near the centre of the measurement vehicle's lateral axis, just behind the windscreen (except the ADMA system).

A total of 583 GNSS tracks were recorded with the GNSS receivers for all road categories combined. The reference lane centrelines were derived from ADMA high-accuracy DGPS measurements.

## Algorithm for determining lane centrelines

In order to successfully derive lane centrelines from the data set, some data pre-processing is necessary. In the first step, a heuristic error filter for finding and excluding potential flawed GNSS positions is applied by using threshold values for error identification, as described by Krampe et al. (2013). In the next step, the actual processing (Figure 3) begins by creating equidistant points (at a pre-set distance) along an OSM road network graph. Depending on the chosen distance, the final output will result in a more or less coarse lane geometry. Perpendiculars which have a total length of 20 m are dropped through these points. These 20 m are divided equally by the road centreline of the input base graph. In a further step, the intersection points between the orthogonal lines created and the GNSS trajectories are

calculated and stored under their respective perpendicular line IDs. Hence, the lateral positions of all vehicle tracks every n metres are obtained.
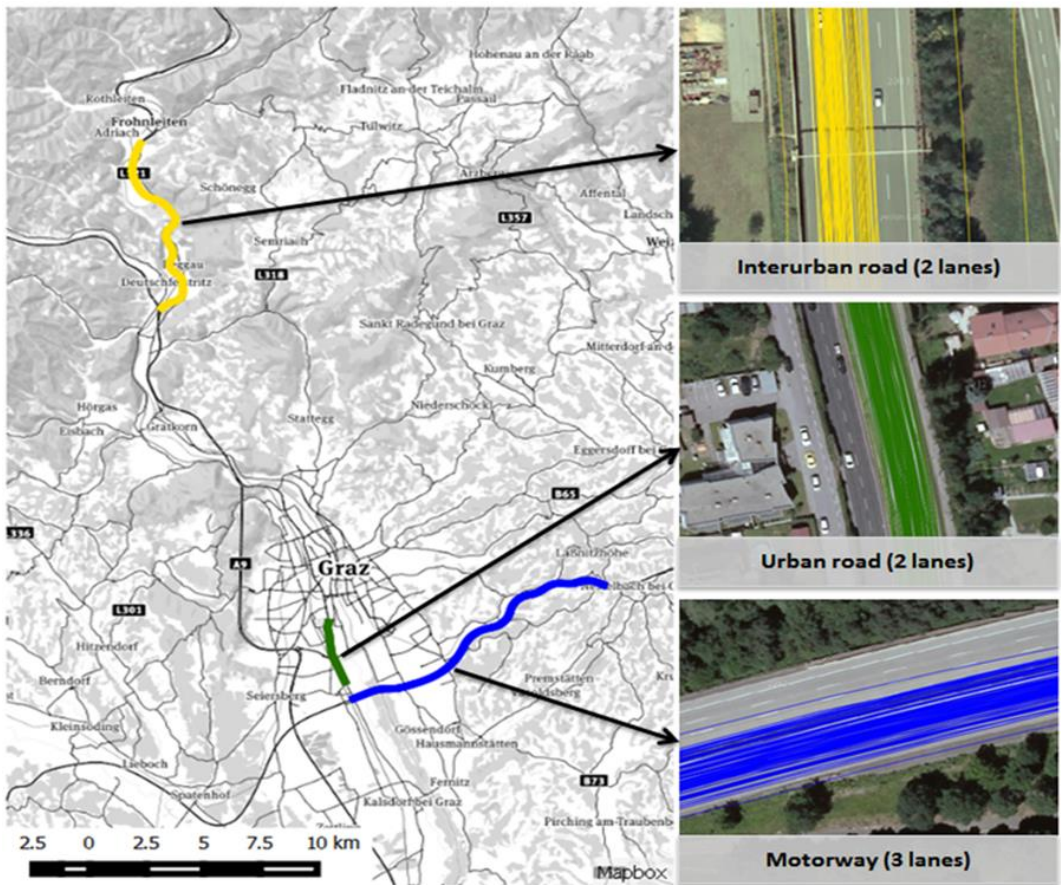


**Figure 2:** Location of three study areas in and around Graz, Austria, with vehicle trajectories shown for one driving direction. Background maps: Mapbox, basemap.at

Those points are again pre-filtered for use in the next step by defining cut-off values for the X-coordinates at specific quantile values, thereby excluding tracks with distinctive systematic deviations from the distribution.

**Figure 3:** Main processing steps for deriving the lane-level road network graph

In order to get a probability distribution of the lateral positions obtained, a KDE is performed on each ID set. The KDE is calculated as follows:

$$\hat{p}(x)\frac{1}{Nh}\sum_{i=1}^{N} K\left(\frac{x-x_i}{h}\right)$$

where K() is the kernel function and h is the bandwidth. Basically, the KDE works as follows: A Gaussian kernel function (in our case curved) is placed over every single input point. For each input point, the kernel function reaches its highest value at just that point and decreases in height (value) with increasing distance from it. By adding up the kernel function values of all data points, the density is estimated. The result of a KDE depends to a very high degree on the smoothing parameter value chosen for the kernel function, the so-called bandwidth. Generally speaking, a small bandwidth leads to a more jagged KDE appearance, resulting from small bias and large variance, while a large bandwidth results in highly smoothed estimates due to high bias and small variance. There are many sophisticated data-driven methods available for choosing the ideal bandwidth. For this study, the "solve-the-equation" plug-in approach developed by Sheather & Jones (1991) was used, which according to Cao & Cuevas (1994) and Jones et al. (1996) performs best on real data-set estimates. However, according to other authors (Bowman & Azzalini, 1997; Soh et al., 2013; Wand & Jones, 1994), the type of kernel function (Gaussian, Epanechnikov, Triangular, Biweight etc.) has little impact on the KDE result.

Once the computation of the KDEs for every set of intersection points is completed, the local maxima of the KDEs are identified through the first and second derivative tests. Based on the assumption that drivers tend to drive in the middle of a lane, it can be concluded that the probability of a vehicle's position is highest along a lane centreline. Therefore, the density of a set of vehicle tracks along a road should be highest at lane centrelines, which manifest themselves as local KDE peaks.

As a result, in the best of cases all peaks found in the KDE should correspond to actual road centrelines. Unfortunately, this is not always the case, since there can be systematically wrong GNSS trajectories within the input data set, caused for instance by bad satellite connection or generally bad GNSS accuracy. These "wrong" estimations are filtered out by means of a distance plausibility assessment. To carry this assessment out, a matrix for the distances between every element in the set of peaks found (PS) is calculated. The result is an n x n two-dimensional array with n x (n-1)/2 distinct values, where n is the number of elements in the PS. Using distance thresholds derived from the reference graph, the PS is filtered for implausible distance relations. Only one peak is allowed within one lane. The found road centrelines need to have a minimum distance between each other, yet must not exceed a maximum distance, which would locate them outside the road. The formula used for calculating the distance matrix is the great circle distance based on the Spherical Law of Cosines, defined as follows:

$$d = \arccos(\sin(lat_1) * \sin(lat_2) + \cos(lat_1) * \cos(lat_2) * \cos(long_2 - long_1)) * R$$

Lat1 and long1 are the latitude and longitude coordinates of the first point, and lat2 and long2 are the coordinates of the second point given in the radian. R stands for the mean earth radius given in kilometres, which varies depending on the ellipsoid used – in this case, 6,371 km for WGS84.

The intersection points between the perpendicular lines and the input GNSS tracks are combined with a distance calculation to estimate the number of lanes at each intersection. Thus, the algorithm output is further improved by narrowing down the number of potential peaks that are found. This is necessary because (as mentioned before) in most cases the number of KDE peaks will not correspond to the exact number of lanes.

In the final step, a graph showing separate-lane geometries is constructed by connecting the remaining adequate peaks of successive perpendiculars, using a shortest-distance algorithm. This final output graph is compared with the reference graph constructed from high-accuracy DGPS measurements.

## 4    Results and Discussion

The method described was tested with data measured on three different road types: urban, interurban and motorway. As Figure 4 illustrates, the methodology can successfully be used for modelling lanes from common GNSS receivers in all of the three environments. The respective KDEs show three peaks for the motorway and two peaks for both the urban and the interurban roads. The methodology thus successfully depicted the right number of lanes.

Furthermore, the KDE peaks manifested themselves not only in correct number, but also in correct position. The X-coordinates of the peaks detected by the algorithm and shown in red are very close to the actual road centrelines represented by the X-coordinates of the reference graph, where they are shown as green line. This test case proves the accuracy with which the methodology developed is able to find the lane positions. The impression is confirmed if the output graph is compared with an orthoimage and the reference graph (see Figure 4).
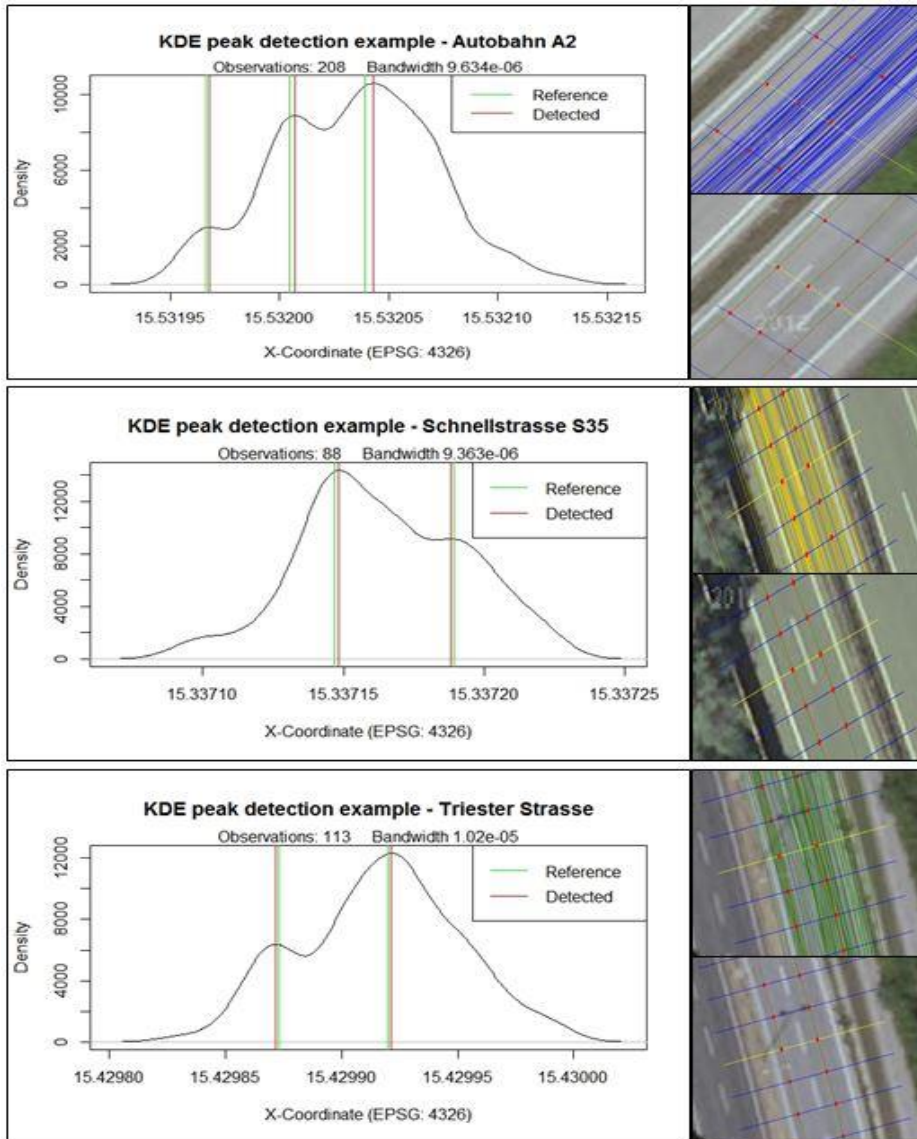


**Figure 4:** Examples for three road types for which lane centrelines were modelled with high accuracy. Background map: basemap.at

The box plot in Figure 5 shows the distance distribution between the estimated centrelines and the reference graph over the complete length of one road for each road type. The result is promising, with median distances of 17 cm for the urban road, 20 cm for the interurban road, and approximately 18 cm for the motorway. Despite this promising result, many distances tend to have higher values, reaching about 1.5 m for the urban and 1.9 m for the interurban road. The greatest distance on the motorway extends to about 1.6 m. Taking into account that lanes are generally between 2.75 m and 3.75 m wide, the algorithm will be able to reliably model the lanes, though with some limits to the absolute accuracy.
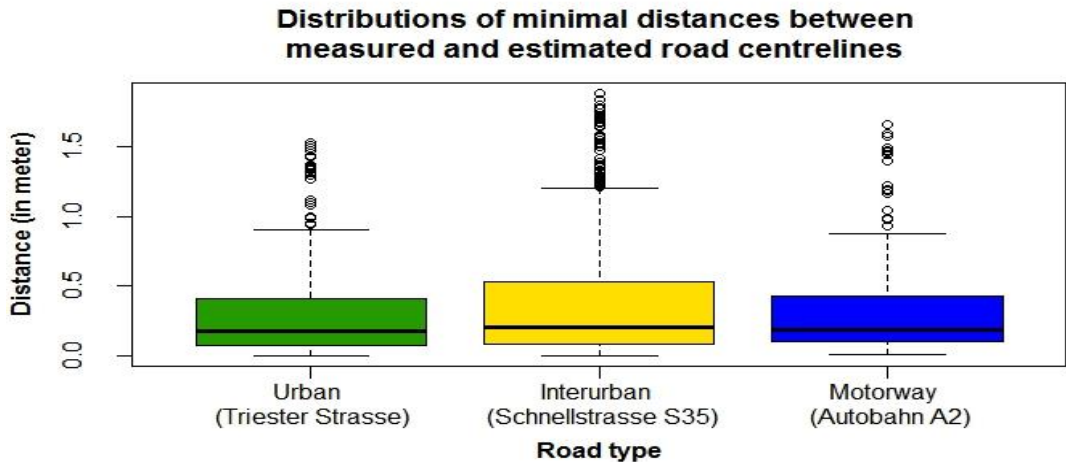


**Figure 5:** Box plot showing distribution of distance deviations between modelled and reference lane centrelines for the three road types

On the other hand, it could be demonstrated that the quality of the output results is highly dependent on the positional precision of the input GNSS trajectories.

In this context, the GNSS tracks are to some degree a representation of driving behaviour. How far they mirror driving behaviour can best be seen if we look at the ADMA GPS trajectories. The distance deviation between the reference graph and the various ADMA measurements can be as much as 60 cm. Since the ADMA system is highly accurate (<3 cm), these disparities are due to differences in driving behaviour: as the measuring car is about 1.8 m wide and a lane on average 3.5 m, the driver can navigate 1 m in either direction of the lane centreline without leaving the lane.

Such systematic distortions lead to wrong peak estimations when using the KDE. This is due to the change in the distribution of the trajectories on the road, which can lead to wrong clustering manifesting itself in the KDEs as incorrect peaks. Positionally inaccurate GNSS tracks can also blur correct bundling of tracks, thus leading to an underestimation in the number of detected lanes. All these cases would violate the assumption that the highest density of vehicle tracks can be found at the lane centrelines, affecting the positional and geometrical accuracy of the methodology.

It should be pointed out, however, that the level of spatial accuracy required also depends on the network graph's use case. The level of accuracy required ranges from just a few centimetres for lane-keeping and autonomous driving applications, to several metres for lane-level traffic information and navigation (EDMap Consortium, 2004; Wilson et al., 1998). Further research could be conducted regarding both the data pre-processing steps and the algorithm itself. Presumably, the most important information needed in order to construct the geometry successfully is the number of lanes. If this parameter could reliably be derived from the data itself, it would greatly improve the search for the correct peaks within the KDE and prevent the algorithm from detecting too many peaks. Moreover, it appears likely that this would allow the selection of smaller bandwidth values, leading to more variance in the KDE through which less pronounced, but correct, bundling of trajectories could be considered. As the data pre-processing is almost as important as the road centreline detection algorithm, one step which could be undertaken is the selective thinning of GNSS tracks with high offset. This could lead to more peak manifestations in KDEs where too high a homogeneity of the input data leads to over-smoothing.

Although the testing of the performance of the different devices is outside the scope of this paper, the author is aware of the potential impact on the analysis results. However, it can be assumed that the algorithm's performance varies according to the quality of the input data. Because of this, determining a lane's centreline will work better the higher the spatial accuracy of the GNSS tracks, and thus the lower the spread of the data. Defining quality specifications for the input data could also be the subject of further research.

Moreover, research is required regarding the applicability of the algorithm developed in this context for complex street settings like road junctions.

## Acknowledgments

## References

Bowman, A. W., & Azzalini, A. (1997). Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations. OUP Oxford.

Cao, R., & Cuevas, A. (1994). A comparative study of several smoothing methods in density estimation. Computational Statistics &amp; Data Analysis, 17(2), 153–176. http://doi.org/10.1016/0167-9473(92)00066-Z

Chen, Y., & Krumm, J. (2010). Probabilistic Modeling of Traffic Lanes from GPS Traces. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (pp. 81–88). New York, NY, USA: ACM. http://doi.org/10.1145/1869790.1869805

Edelkamp, S., & Schrödl, S. (2003). Route planning and map inference with global positioning traces. In Computer Science in Perspective (pp. 128–151). Springer.

EDMap Consortium. (2004). Enhanced digital mapping project. Final report. EDMap Project eSafety Forum.(2005). Digital maps Working Group Final Report. European Commission (eSafety Forum), Brussels.

Faouzi, N.-E. E., Leung, H., & Kurian, A. (2011). Data fusion in intelligent transportation systems: Progress and challenges – A survey. Information Fusion, 12(1), 4–10. http://doi.org/10.1016/j.inffus.2010.06.001

Gomes, L. (2014, August 28). Hidden Obstacles for Google's Self-Driving Cars. Retrieved 26 January 2016, from http://www.technologyreview.com/news/530276/hidden-obstacles-for-googles-self-driving-cars/

Jones, M. C., Marron, J. S., & Sheather, S. J. (1996). A Brief Survey of Bandwidth Selection for Density Estimation. Journal of the American Statistical Association, 91(433), 401–407. http://doi.org/10.1080/01621459.1996.10476701

Krampe, S. (2007). Nutzung von 'Floating Traveller Data (FTD)' für mobile Lotsendienste im Verkehr. Techn. Univ, Darmstadt.

Krampe, S., Pfister, J., & Pucher, G. (2013). IMPAKT – GPS Assessment for Cooperative Services (Car2x-Communication). In Proceedings mobil.TUM 2013 – ITS for Connected Mobility. Munich.

Lu, M., Wevers, K., & Van Der Heijden, R. (2005). Technical feasibility of advanced driver assistance systems (ADAS) for road traffic safety. Transportation Planning and Technology, 28(3), 167–187.

Pandazis, J.-C. (2002). NextMAP for Transport Telematics Applications. project IST-1999-11206, final report.

Sheather, S. J., & Jones, M. C. (1991). A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation. Journal of the Royal Statistical Society. Series B (Methodological), 53(3), 683–690.

Soh, Y., Hae, Y., Mehmood, A., Hadi Ashraf, R., & Kim, I. (2013). Performance Evaluation of Various Functions for Kernel Density Estimation. Open Journal of Applied Sciences, 03(01), 58–64. http://doi.org/10.4236/ojapps.2013.31B012

Thomson, R. C., & Richardson, D. E. (1995). A graph theory approach to road network generalisation. In Proceeding of the 17th International Cartographic Conference (pp. 1871–1880).

Uduwaragoda, E. R. I. A. C. M., Perera, A. S., & Dias, S. A. D. (2013). Generating lane level road data from vehicle trajectories using Kernel Density Estimation (pp. 384–391). IEEE. http://doi.org/10.1109/ITSC.2013.6728262

Wand, M. P., & Jones, M. C. (1994). Kernel Smoothing. CRC Press.

Weibel, R., & Jones, C. B. (1998). Computational perspectives on map generalization. GeoInformatica, 2(4), 307–314.

Wilson, C. K. H., Rogers, S., & Weisenburger, S. (1998). The potential of precision maps in intelligent vehicles. Proceedings of the 1998 IEEE International Conference on Intelligent Vehicles., 2.

Zhang, L., Thiemann, F., & Sester, M. (2010). Integration of GPS traces with road map. In Proceedings of the second international workshop on computational transportation science (pp. 17–22). ACM.