The Future of Automotive Localization Algorithms: Available, reliable, and scalable localization: Anywhere and anytime

Rickard Karlsson and Fredrik Gustafsson

Journal Article

Rickard Karlsson and Fredrik Gustafsson, The Future of Automotive Localization Algorithms: Available, reliable, and scalable localization: Anywhere and anytime, IEEE signal processing magazine (Print), 2017. 34(2), pp.60-69.
http://dx.doi.org/10.1109/MSP.2016.2637418
Postprint available at: Linköping University Electronic Press

http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-135786

The Future of Automotive Localization Algorithms

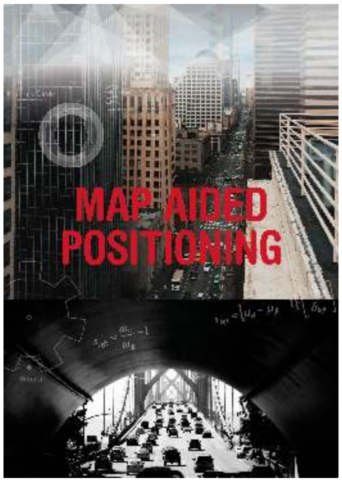
Rickard Karlsson and Fredrik Gustafsson
Department of Electrical Engineering
Linköping University, Linköping Sweden
E-mail: {rickard, fredrik}@isy.liu.se

***Abstract*—Most navigation systems today rely on *global navigation satellite systems* (GNSS), including navigation in cars. With support from odometry and inertial sensors, this is a sufficiently accurate and robust solution, but there are future demands. Autonomous cars require higher accuracy and integrity. Using the car as a sensor probe for road conditions in cloud-based services also sets other kind of requirements. The Internet of Things concept require stand-alone solutions without access to vehicle data. Our vision is a future with both in-vehicle localization algorithms and after market products, where the position is computed with high accuracy in GNSS-denied environments. We present a localization approach based on a prior that vehicles spend most time on the road, with odometer as the primary input. When wheel speeds are not available, we present an approach solely based on inertial sensors, which also can be used as a speedometer. The map information is included in a Bayesian setting using the particle filter, rather than standard map matching. In extensive experiments the performance without GNSS is shown to have basically the same quality as utilizing a GNSS sensor. Several topics are treated: virtual measurements, dead reckoning, inertial sensor information, indoor-positioning, off-road driving, and multi-level positioning.**

***Index Terms*—Localization, map-aided positioning, particle filter, velocity estimation.**

I. Introduction

Today's positioning systems are intended for humans rather than machines. The position is presented and used for instructions in navigation systems, or for reporting vehicle data, also including emergency accident location. We refer to the area as *localization algorithms* for several reasons. First, the word *algorithm* indicates software development. Already today there is sufficient information at hand, in terms of sensors and databases to make a leap in performance compared to GNSS-based solutions. Second, localization is not a *system*, it is rather a service required by many systems. Third, the term *navigation* is avoided since this is only one application of localization algorithms. Fourth, localization is sometimes a more appropriate term than positioning, since a true longitude and latitude position is of no value unless the map and situational awareness have the same absolute accuracy.



Consider the schematic picture of a vehicle in Fig. 1. The trend is to make vehicles autonomous, [1–4] and utilizing *advanced driver assistance systems* (ADAS). Hence, there is a need to improve both localization and velocity estimation systems. Basically going beyond traditional point estimation methods, [5, 6] to get a better probabilistic understanding [7–10] of the environment using more detailed models and filters. The actuators (brake, steering wheel, engine torque) have basically been the same since the car was invented, and only a few new actuator concepts have been introduced (active suspension, movable headlights, *etc*).

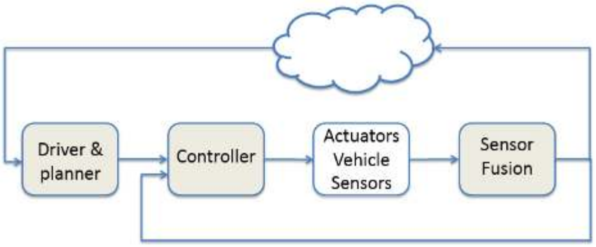


Fig. 1: Illustration of data flow in a vehicle. Future ADAS functionality might include cloud information as well as control, sensor fusion, and planning.

In stark contrast to the actuators, the amount of sensors has increased substantially over the last decade, [11–13], for example:

- *Inertial measurement unit* (IMU), [14], in the *engine control unit* (ECU) and in suspension sensors for estimating the vehicle state.
- Vision, stereo vision, night vision, radar, sonar for monitoring the surroundings and keeping the vehicle in the lane at a safe distance (*i.e.*, relative position control).
- The *wheel speed sensors* (WSS) introduced with the ABS systems are one of the most versatile sensors in the car.
- Databases: vectorized road maps, [15–18], utilized for positioning including road height, map-matching, [11, 19–22], and pothole indications, [23], etc.
- Cars are slowly following the development of smart phones. Already today there are many radio receivers in the vehicles: cellular network, Bluetooth, and Wi-Fi. These can be used in various signal processing applications such as localization and speed estimation.

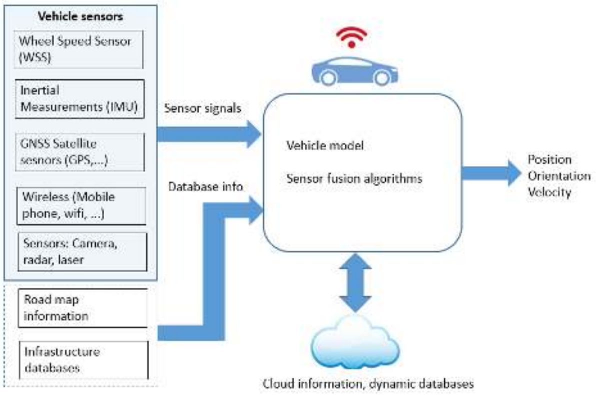


Fig. 2: Overview of positioning, orientation, and velocity estimation utilizing all available in-vehicle sensors, external databases, and cloud interaction.

It is less explored that these information sources all include indirect information about the position. The vehicle state sensors contain information of road signatures (curves, banking, slopes and small variations in the surface height). The vision sensors can see landmarks of known position. How the WSS can be used for positioning will be thoroughly described below.

Sensor fusion is used in all cases above for refining the information, where there are several good examples of virtual (or soft) sensors that compute physical quantities that cannot readily be measured by sensors. Examples include detection of obstacles, pedestrians, animals from vision sensors, and tire pressure and road friction from WSS. The approach we propose is based on statistical signal processing techniques, based on a simple odometric model of the vehicle and a model of each sensor relating to the vehicle state. In particular the road map information is nonlinear and cannot be approximated with a linear Gaussian model, so a particle filter framework is preferred to Kalman filter algorithms. The sensor fusion concept is summarized in Fig. 2.

The paper is organized as follows. In Section II different localization application areas are highlighted. In Section III map matching is addressed. Section IV addresses dead reckoning. Section V introduces Bayesian estimation for GPS free positioning. In Section VI inertial sensors are discussed and utilized. Finally, Section VII summarizes the contribution and discusses further ideas.

II. Future Localization Algorithm Applications

This section motivates the need for improved localization algorithms highlighting areas such as cloud based computations, autonomous driving, hand-held devices, and mapping.

A. Cloud-based Services

To some extent, positioning is used for cloud based crowd sourcing today, such as in apps for pothole detection, speed camera positions, *etc*. This is an area that most probably will explode in the future, when the manufacturers integrate these reports in their own servers, and offer their own and other customers services based on this information.

As an example of a virtual sensor, consider potholes that are annoying to passengers and may be a hazard to the vehicle. These are easily detected by accelerometers, WSS or suspension sensors and the presence of potholes can be included in the car's navigation system. The problem is how to share the information between users. Fig. 3 shows an illustration of pot hole detections and clustering, [23]. Many vehicles have in this case hit the same potholes, and delivered the estimated position to a cloud database. A cloud based clustering algorithm is then used to merge them into one pothole, and possibly also project the position to the road. This information can now be shared with other drivers, but it could also be used by road authorities for maintenance.

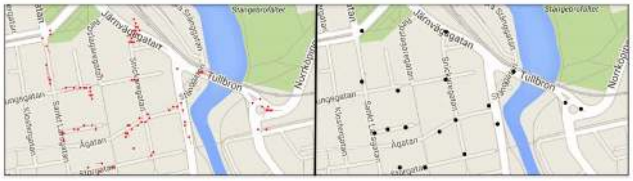


Fig. 3: There are different sensors in a modern vehicle that can detect potholes. Reports from a fleet of vehicles from NIRA Dynamics are sent to a cloud database (left fig), [23]. Since their notion of position differs for natural reasons (based on GNSS), clustering is needed in the cloud (right fig).

B. Autonomy

Future autonomy will put high demands on the localization algorithms. Despite the media success of self-driving cars, the technology is still in development. On one hand, there is the DARPA generation of cars where localization is based on a laser scanner, whose price is still far from affordable, See DARPA Grand challenge and urban challenge [1], [2]. Further, its raised placement on the rooftop is not well aligned with the design. The Google self-driving car, [3] is equipped with laser, radar, and cameras on the roof top. Apart from most vehicles it is not designed or even possible to drive manually. On the other hand, we have seen self-driving cars (Audi RS7 Piloted Driving presented at Hockenheim) positioned using differential GPS, including yaw estimation from multiple antennas, and camera information. These cars have demonstrated spectacular performance on restricted accurately mapped areas.

Autonomous functions in the car, and in the extreme self-driving cars, will need another level of integrity. The localization algorithms must work in tunnels, parking houses, urban canyons, and other areas where GNSS has problems. If the satellite signal is only mitigated and pseudo-ranges are available multiple model filters and map constraints might be an option, [24]. However, for the general case: indoor driving, long tunnels, and multiple levels, the focus is on map aided positioning without satellite signals.

Localization must also be robust against jamming and spoofing. Fleet management and theft tracking systems should not rely on the access to GNSS.

C. Devices

With tens of billions of connected devices around us, some of them are to be used in vehicles. There is of course a demand to keep track of those devices. In some cases, the devices can be connected to the car to take advantage of the sensor information on the CAN (*computer area network*) bus. However, there is no standard for the protocol here, so making devices connect to many OEM (*original equipment manufacture*) vehicles is a challenge. That is, there is also a need for completely stand-alone localization algorithms. It will also be more common with transition from in-vehicle estimation to hand-held devices.

D. Mapping

For positioning/localization there could either be a priori map information available or it could be derived from sensor data. For vehicle positioning usually accurate vectorized maps of the road network are available. This is the focus on the applications described in this paper. Also many landmarks such as speed signs etc could be considered known and available in complementary databases.

For the sake of completeness and since the algorithms and methods are closely related we will briefly discuss *Simultaneous localization and mapping* (SLAM). It is an extension of the localization problem to the case where the environment is unmodeled and has to be mapped on-line. A survey to the SLAM problem is given in [25–27]. The FastSLAM algorithm introduced in [28] has proved to be an enabling technology for such applications. FastSLAM can be seen as a special case of *Rao-Blackwellized particle filter* (RBPF) or *marginalized particle filter* (MPF) (see Section V-A), where the map state containing the positions for all landmarks used in the mapping can be interpreted as a linear Gaussian state. The main difference is that the map vector is a constant parameter with a dimension increasing over time, rather than a time-varying state with a dynamic evolution over time. In [29] SLAM is used to get high accuracy map information (centimeter resolution) utilizing all available sensors such as GPS, odometer, and laser. In [30] a different technique utilizing image data for high accuracy navigation is utilized.

III. Road Maps and Map Matching

The unique feature with automotive localization algorithms is that vehicles spend most of their time on roads, and this is also the common theme in this paper. We will illustrate how road maps can be integrated with sensor fusion techniques to provide an accurate position with high integrity.

The classical method to improve localization performance is *map matching*, [22, 31]. Here, the position estimate computed from the sensors (for instance GPS) is mapped to the closest point on the road. This is an appropriate method for presentation purposes, but it suffers from two problems. First, it does not take the topography of the map into account, which implies that the localization can jump from one road to another. Second, the motion dynamics of the vehicle is not combined with the map information in an optimal way. Having said that there are a different types of map matching, basically using the estimated trajectory in combination with the GPS measurement in order to retrieve the most likely position. They are sometimes referred to as point-to-point, point-to-curve, and curve-to-curve matching, [11, 19–21]. For a detailed survey over map-matching we refer to [22].

The purpose of this section is to survey different methods which we refer as *dynamic map matching*. This includes combining a motion model, sensor models and the road model in a nonlinear filter, including uncertainties. The problem is basically to fit a distorted and noisy trajectory to the road network. Fig. 4 illustrates the principle. Hence, it is possible to utilize only odometry and map information to get an accurate localization, [32, 33].

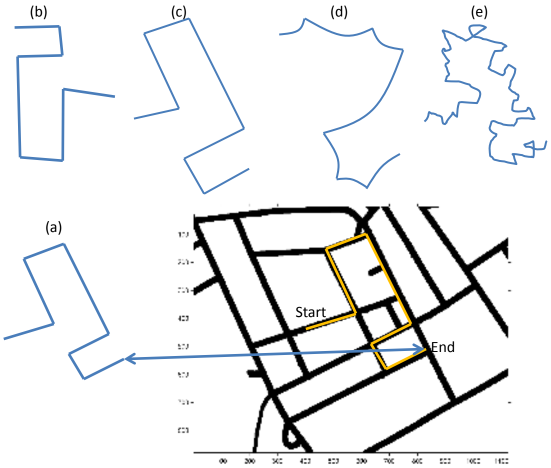


Fig. 4: The key idea in dynamic map matching is to fit an observed trajectory to the road network. (a) Undistorted trajectory. (b) Undistorted trajectory with random rotation. (c) Trajectory based on biased speed. (d) Trajectory based on biased yaw rate. (e) Trajectory with random noise.

IV. Dead-Reckoning Principles

Dead reckoning is basically calculating the integral of velocity or acceleration signals, with or without a vehicle model. It can be based on IMU data or WSS signals for instance. We will look at several aspects:

- Odometry or dead reckoning based on WSS
- Inertial sensor data dead reckoning
- Utilization of map matching
- Dynamic filtering

A. Dead-reckoning: Odometry

A simple motion model is based on a state vector consisting of position X, Y and course ψ, in which case the principle of *dead-reckoning* can be applied. Inserting the observed speed $\vartheta^m(t)$ and angular velocity $\dot{\psi}^m(t)$ as input signals gives the following dynamic model with process noise $w(t)$:

$$\begin{aligned}
X(t+T) &= X(t) + \vartheta^m(t)T\cos(\psi(t)) + T\cos(\psi(t))w_\vartheta(t),\\
Y(t+T) &= Y(t) + \vartheta^m(t)T\sin(\psi(t)) + T\sin(\psi(t))w_\vartheta(t),\\
\psi(t+T) &= \psi(t) + T\dot{\psi}^m(t) + Tw_{\dot{\psi}}(t).
\end{aligned}$$

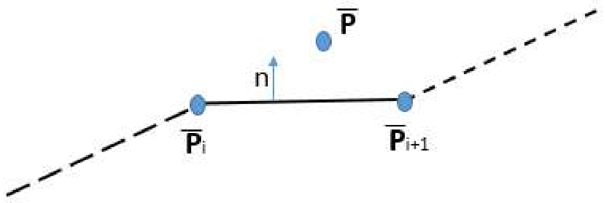


Fig. 5: Road segment i with normal vector $\hat{n}$ and a position estimate $\bar{p}$. The scalar product between the vectors can be used in order to determine if the estimate should be considered to belong to the segment or not.

This model has the following structure ($T = 1$):

$$x_{t+1} = f(x_t, u_t) + g(x_t, u_t)w_t, \qquad u_t = \left(\vartheta_t^m, \dot{\psi}_t^m\right)^T.$$

Normally, additional sensors are needed to get observability of the absolute position. However, the vectorized road map, [15–18], contains sufficient information in itself. Note that the speed and the angular velocity measurements are modeled as inputs, rather than measurements. This is in accordance to many navigation systems, where inertial measurements are dead-reckoned in similar ways. Alternative road graph models are discussed in [34], and second order motion models in [35].

B. Dead-reckoning: Inertial Sensors

Using an IMU it is possible to measure the acceleration and angular rotation directly. From these measurements it is theoretically possible to integrate the underlying system to achieve position, velocity and direction, [14]. This is a common approach for military aircraft navigation and underwater navigation. It is possible to use this dead-reckoning together with map-matching to mitigate sensor imperfections. This is probably best achieved using dynamic filtering and will be described more in the sequel. For cheap commercial sensors usually a GPS sensor is needed to handle the drift due to small sensor biases.

C. Dead-reckoning: Map matching

As discussed previously, map-matching can be done by fitting an estimate to the closest road or by looking at segments etc. Here we will focus on the point-to-point matching, *i.e.*, that the estimate is mapped to the closest orthogonal distance. In Fig. 5 a position estimate ($\bar{p}$) is considered to belong to road segment i, *i.e.*, between the two road edges $\bar{p}_i$ and $\bar{p}_{i+1}$. This can easily be verified if the following scalar products are greater than zero: $\bar{p}_i \cdot \Delta_i > 0$ and $\bar{p}_{i+1} \cdot (-\Delta_i) > 0$, where $\Delta_i = \bar{p}_{i+1} - \bar{p}_i = (dX, dY)^T$. If this is the case the closest distance to the segment can be calculated as $d = ||(\bar{p} - \bar{p}_i) \cdot \hat{n}||$, using the normal vector $\hat{n} = (dY, -dX)^T$. Such project is needed in many localization systems and in particular utilized in the method described in Section V.

D. Dead-reckoning: Dynamic Filtering

A generic nonlinear filter for localization consists of the following main steps:

- *Time update :* Use a motion model to predict where the vehicle will be when the next measurement arrives.
- *Measurement update:* Use the current measurement and a sensor model to update the information about the current location.

In a Bayesian framework, the information is represented by the posterior distribution given all available measurements. The process of computing the Bayesian posterior distribution is called filtering. Details are given in Section V, where the distance calculated in Section IV-C is used in a probabilistic way.

V. Bayesian Filtering for Map Aided Positioning

A. Bayesian Filtering

Nonlinear filtering is the branch of statistical signal processing concerned with recursively estimating the state x_t based on the measurements up to time t, $\mathbb{Y}_t \triangleq \{y_1, \ldots, y_t\}$ from sample 1 to t. The most general problem it solves is to compute the Bayesian conditional posterior density $p(x_t|\mathbb{Y}_t)$. There are several algorithms for computing the posterior density. The *Kalman filter* (KF) [5] solves the filtering problem in case the model is linear and Gaussian. The solution involves propagating the mean $\hat{x}_{t|t}$ and the covariance $P_{t|t}$ for the posterior distribution. For nonlinear problems the model can be linearized before the KF technique is applied, leading to the *extended Kalman filter* (EKF), [6]. There also exist methods where the Gaussian approximation is the key element, hence no linearization is needed, for instance the *unscented Kalman filter* (UKF) [36] approximates the posterior at each step with a Gaussian density. Common for these methods is that it is not trivial to impose hard-constraints from the road-map. They also do not work particularly well unless the posterior density is very mono-modal or Gaussian. For a KF based estimation with map information see [37].

The road constraints imply a kind of information that normally leads to a multi-modal posterior density (the target can be on either this road, or that road, etc). Hence, a Gaussian approximation of the PDF is not suitable. A completely different approach to nonlinear filtering is based on approximating the posterior $p(x_t|\mathbb{Y}_t)$ numerically. The *point mass filter* (PMF) [38] represents the state space using a regular grid of size N, where the grid points and the related weights $(x^{(i)}, w_t^{(i)}$ are used as a representation of the posterior. Different basis functions have been suggested, the simplest one being an impulse at each grid, when the posterior approximation can be written $p(x_t|\mathbb{Y}_t) \approx \sum_{i=1}^{N} w_t^{(i)} \delta(x_t - x_t^{(i)})$, where $\delta(x)$ denotes the Dirac-delta function. The *particle filter* (PF) [10] is the state of the art numerical solution today. It uses a stochastic grid $\{w_t^{(i)}, x_t^{(i)}\}_{i=1}^{N}$ that automatically changes at each iteration.

Depending on the model it is also possible to implement numerical efficient filters combining KF and PF. The idea is to divide the state space into two parts. If there is a conditionally linear Gaussian substructure with this partition, the KF can be utilized for that part and the PF for the other part. This is referred to as the *Rao-Blackwellized particle filter* (RBPF) or the *marginalized particle filter* (MPF), [7–9, 39–41] The RBPF improves the performance when a linear Gaussian substructure is present, *e.g.*, in various map based positioning applications and target tracking applications as shown in [41].

The map aided positioning algorithm based on the particle filter is summarized in Alg. 1.

Algorithm 1 Particle Filter for Map Aided Positioning

Given the system

$$\begin{aligned} x_{t+1} &= f(x_t) + w_t \\ y_t &= h(x_t) + e_t \end{aligned}$$

```
1: Initialization: For $i = 1, \dots, N$, $x_{0|-1} \sim p_{x_0^{(i)}}(x_0)$ and set $t = 0$.
2: PF measurement update: For $i = 1, \dots, N$, evaluate the importance weights $\tilde{\omega}_t^{(i)} = p(y_t | x_{t|t}^{(i)}, \mathbb{Y}_{t-1})$, and normalize $\omega_t^{(i)} = \tilde{\omega}_t^{(i)} / \sum_j \tilde{\omega}_t^{(j)}$ using map information.
3: Resample $N$ particles with replacement:
   $\Pr(x_{t|t}^{(i)} = x_{t|t-1}^{(j)}) = \omega_t^{(j)}$.
3: PF time update: For $i = 1, \dots, N$ predict new particles $x_{t+1|t}^{(i)} \sim p(x_{t+1|t} | \mathbb{X}_t^{(i)}, \mathbb{Y}_t)$.
4: Increase time and repeat from step 2.
```

B. Particle Filter Based Map Aided Positioning

In this section the *map aided positioning* (MAP) method is first illustrated on experimental data. After that the crucial map based observation is described in detail. Finally, the algorithm performance is presented on 10 experiments conducted in the same driving scenario (see Fig. 7).

1) MAP illustrations: In Fig. 6 the map aided positioning using wheel speed information and road map information is demonstrated, where GPS information is used as a ground truth reference only. For other map aided positioning applications see for instance [42–48]. First the PF is initialized in the vicinity of the GPS position. The initial distribution is chosen uniformly on road segments in a region around the GPS fix. Particles are allowed slightly off-road to handle off-road situations and small map errors. In Fig. 6 (a) the algorithm has been active for some time. As seen, the PDF is highly multi-modal (several clusters of particles). Note that the PF algorithm uses only wheel speeds from the CAN-bus and that the GPS is only used to evaluate the ground truth. In Fig. 6 (b), after yet some turns, the filter has converged and the mean estimate (red circle) is close to the true position (blue circle).

2) MAP algorithm: As discussed in Section IV-C map matching can be used to fit an estimate to the closest road segment. In this section we will focus on the PF implementation, so for each particle it is crucial to find the closest road segment. The generic PF algorithm is used for map aided positioning. The road map is used as a virtual sensor, so there is not an actual measurement function. Instead the closest distance to every road segment is evaluated for each particle. The main advantage here compared to normal map matching algorithms is that the entire probability density is considered, not just one point estimate. In Fig. 5 a particle ($\bar{p}$) is considered to belong to road segment i, if the two defined scalar products are positive (see Section IV-C). The calculated distance can then be assumed Gaussian distributed and used in the PF measurement update.

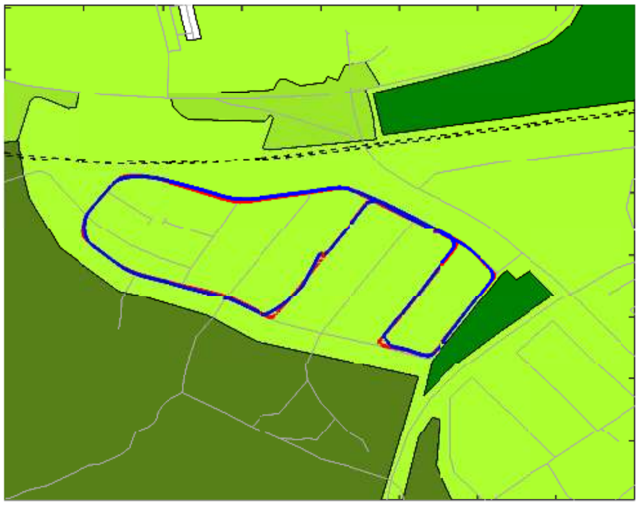


Fig. 7: Ten routes were driven (red GPS line). For one of them (blue line) the position using map aided position is depicted.

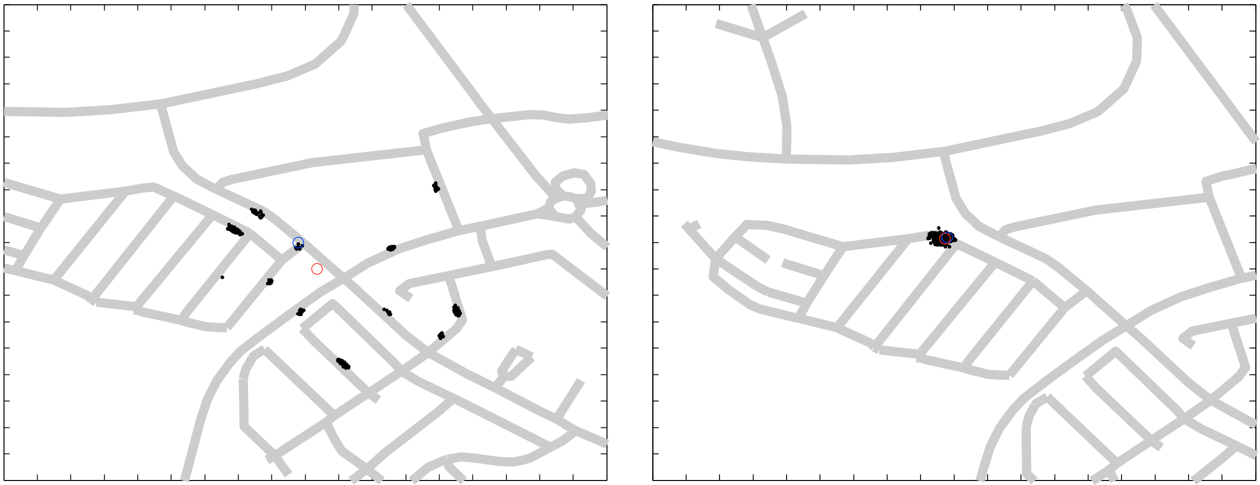
In order to make the algorithm efficient each particle will remember the road segment that was closest in the last update. Hence, if the distance is still close enough not every road segment needs to be considered.

3) MAP performance: To evaluate the average performance 10 similar experiments were conducted (see Fig. 7). All trajectories are driven approximately in the same way, however it is not possible to have them synchronized in time. Hence, RMSE evaluation is done at some fixed way-points. In Fig. 8 the RMSE is calculated for each trajectory against known GPS-way-points (upper plot). In the lower plot the average RMSE is presented. As seen the performance is comparable to standard GPS position error.

C. Multi-Level Positioning and Indoor Navigation

Indoor navigation is challenging since no GPS signal is available. However, for parking garages the map aided positioning principle can still be utilized if maps are available. These maps can also include obstacles such as pillars, side-walls etc defining boundary regions. Since the geometry and curvature are quite different than when driving on normal roads, it is also beneficial to extend the simple vehicle model and include a more precise spatial vehicle model. This can easily be done by evaluating side-walls and pillars around a rectangle placed around the particles which corresponds to the vehicle's geometry.

In order to handle multi-levels one successful implementation is to utilize entry and exit points between levels, and simply handle each level as a continuation of the previous. Inertial sensors such as inclination in longitudinal accelerom-



(a) Illustration of the particle cloud after some iterations. Multi-modal PDF representing the position (several clusters of particles). The particles clustered but still the mean point estimate (red circle) does not correspond to the ground truth GPS (blue circle).

(b) The PF has converged to a uni-modal PDF (one cluster). The mean estimate is now close to the ground truth.

Fig. 6: Map aided positioning using wheel speed sensor information in combination with road map information. The small black dots are particles, the red and blue circles represent mean estimate and ground truth (GPS) position. The two figures represent time instances after initialization when the filter is still in a multi-modal state (a), and at convergence (b).

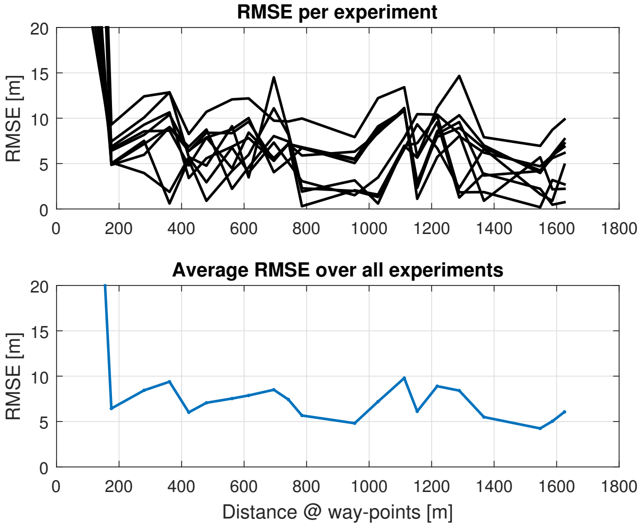


Fig. 8: RMSE for each trajectory as a function of driven distance evaluated at specified way-points and as a total RMSE average at the same instances. Note that GPS was not used in the estimations, but only as a source for ground truth evaluation.

eter signal can also support the level change decision. Similar geometries can be found in multi-level highways in big cities.

For the positioning to work well on highways there must be some variation in the geometry. For instance driving on very straight highways for a long time will of course lead to a more uncertain position estimate, since the wheel radius cannot be known or estimated exactly. However, for such scenarios it is very likely that GPS reception is very good and could be used to adjust the position estimate.

D. Measurement Features

The PF method for positioning is very general, and it is easy to add other information sources to the measurement update. There are several important and common information sources for automotive positioning that can be utilized. Modern camera based vehicles can be equipped with traffic sign recognition etc. If a database of the sign location is available it fits perfectly into the map aided positioning framework, by simply adding a measurement update. Other information sources such as maximum allowed speed information for a road can be used for making a better proposal in the particle filter. In the future more localization sources will probably be available, for instance information exchange from the infrastructure.

VI. Inertial Sensors and Velocity Estimation

In this section the use of inertial sensors such as accelerometers and gyros, both for dead reckoning, sensor fusion, and as stand-alone velocity estimation is discussed.

A. Inertial sensors

An IMU measures acceleration and angular rotation. From these measurements it is possible to integrate the underlying system to achieve an estimate of position, velocity and direction, [14].

Dead-reckoning IMU: The pure integration or dead-reckoning approach relies on very accurate and expensive sensors that are not possible to utilize in commercial passenger vehicles. The main problem for affordable commercial sensors is that they have both unknown sensor biases and drifts, that

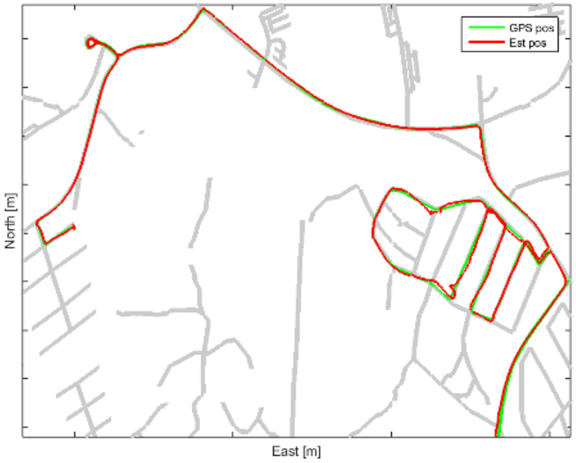


Fig. 9: The fused position EKF estimate and the GPS position depicted in a street map utilizing sensor data from a Google Nexus Android mobile phone.

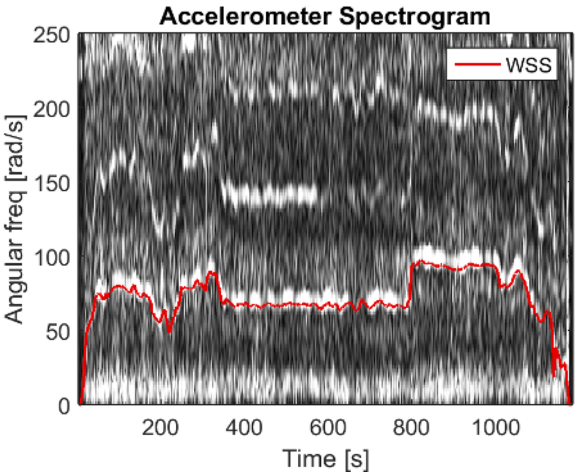


Fig. 10: Accelerometer spectrogram (filtered periodogram of the lateral acceleration as a function of time) compared to the velocity from WSS, where the harmonics correspond to the wheel rotation frequency.

are not possible to remove. For instance in the longitudinal acceleration direction both sensor errors and hills will act like an unknown time-varying bias. Integrating twice to achieve position will very rapidly yield large position errors. For reliable stand-alone navigation without relying on for instance GPS this is very difficult.

Map aided positioning using IMU: An IMU based map aided position (stand-alone or in combination with WSS sensors) was tested in [46], where the outcome was that WSS is a superior velocity sensor due to the above mentioned problems.

GPS *and* IMU *fusion*: For modern ECU:s the in-vehicle sensor cluster consisting of rate gyros and accelerometers can be used together with a GPS sensor to achieve position, velocity, and orientation estimates, [14]. However, not all vehicles have these signals or they are not easily available. Hence, for many application external sensors might be interesting, particularly utilizing signals available in smart phones.

Consider the following state vector

$$x = \begin{pmatrix} q & p^i & v^b & a^b \end{pmatrix}^T,$$

where q is the quaternion vector, p^i is the inertial position vector, v^b is the velocity in the body system, and a^b is the accelerometer vector in the body system. Let C_{ib} be the conversion matrix from body to inertial systems. Hence, in discrete time we have the inertial position $p^i_{t+1} = p^i_t + TC_{ib}v^b_t$. It can be shown that

$$\dot{q_{bi}} = \frac{1}{2}S(\omega^b_{bi})q_{bi},$$

where ω^b_{bi} is the angular velocity of the body system relative to the inertial system described in body coordinates.

In Fig. 9 the EKF estimate based on IMU and GPS signals from a Google Nexus mobile phone using the data logging from [49] is depicted together with GPS position using a discretized model. For this application it is essential that some velocity estimation algorithm or position sensor is utilized to mitigate the dead-reckoning problem with unknown biases in accelerometer and gyro signals. Here the GPS sensor was utilized instead of map-matching techniques.

B. *Virtual Speedometer*

For some applications an accurate velocity estimate is essential. A complementary method to the previously described estimation is to utilize frequency analysis. It turns out that the velocity of the vehicle is proportional to vibrations in the accelerometer signal, [50]. This can be utilized in the previously described positioning filter. It is not as accurate as WSS information, but for a stand-alone application when WSS is not available, data from a standard smart phone can be utilized. In Fig. 10 the lateral accelerometer spectrogram is depicted together with the angular velocity from the WSS (as ground truth). The spectrum is formed at every instance (downsampled to every second) by filtering the periodogram of the lateral acceleration. As seen there are usually a frequency related to the velocity. Also note that there is usually some overtones as well. Utilizing this in the frequency domain, it is possible to construct a simple peak detector to estimate the velocity. In Fig. 11 the point estimates utilizing only the maximal peak in the accelerometer spectrum (batch-wise every second) is depicted. As seen usually this correlates to the true velocity, but there are some outliers. It is possible to improve the detector by incorporating the knowledge of overtones but this is not done here.

For map aided positioning WSS sensor information is always available. But for stand-alone applications the vibration based velocity estimate could be used as a complementary measurements for map-aided positioning. In Fig. 12 it is illustrated how the speed estimation can be used as input signal (instead of the WSS data) to the map aided positioning. For low velocities this is not accurate enough, but potentially pure accelerometer integration can be used for short period of times to support the algorithm.

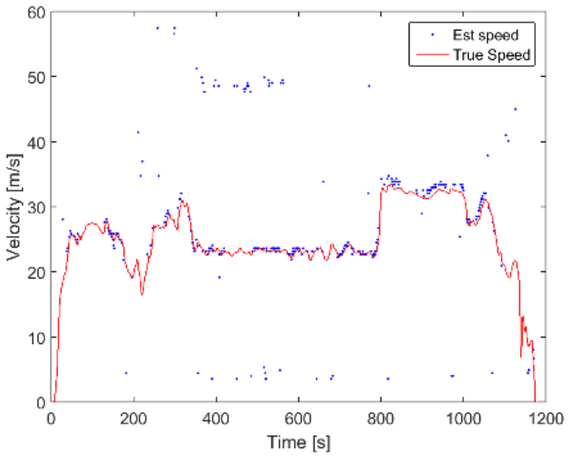


Fig. 11: Velocity from WSS compared to estimates derived from peaks in the accelerometer spectrum. Sometimes there are outliers or the wrong harmonic is selected, however this can later be corrected by the over-all velocity filter.

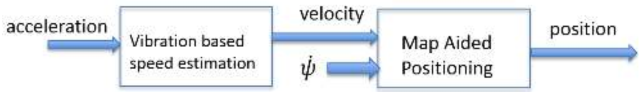


Fig. 12: The general idea for a stand-alone map aided positioning when WSS is not available is to use the acceleration vibration speed estimation as input to the MAP algorithm. With proper outlier rejection to the velocity estimates the algorithm can function without WSS data.

VII. Discussion and Conclusion

We have discussed the needs in future automotive localization algorithms, and pointed out that both accuracy and integrity have to be improved compared to the navigation systems today that rely on GNSS. For this purpose, we outlined a path to future automotive localization algorithms based on a statistical signal processing approach, where information from various sensors and information sources are fused based on given sensor models and an odometric motion model. The possible sensors include in-vehicle sensors such as wheel speed sensors, accelerometers, gyros, and external ones such as GPS. However, localization concerns the relative position of the own vehicle compared to the surrounding, so the position relative the road network is more important that the absolute longitude and latitude. A road map is the key information source for this purpose, and we have discussed the concepts of map matching (basically projection of a position to the road network) and map aided positioning (where the road map is treated as a sensor). Furthermore, landmarks such as road signs detected by a camera, and the inclusion of car to infrastructure information, wireless sources (Bluetooth, Wi-Fi, and mobile positioning) will be crucial in the future, and this information is also easily incorporated in our framework.

We have in particular highlighted the crucial concept of map aided positioning. Utilizing measurements from a yaw rate and wheel speed signals (WSS), we have shown that it is possible in urban areas to position a vehicle with almost GPS accuracy without using any external GNSS positioning sensor, when utilizing the vehicle model and accurate road map information in a particle filter. Incorporation of inertial sensor measurements for velocity estimation utilizing accelerometer vibrations was demonstrated, as well as the basic principle when using it for positioning.

Acknowledgment

This work was partially supported by the Wallenberg Autonomous Systems Program (WASP).

Author Bibliography

R. Karlsson received the M.Sc. degree in Applied Physics and Electrical Engineering in 1996, and a Ph.D in Automatic Control in 2005, both from Linköping university, Sweden. He became an associate Professor in 2010. He has worked with automotive signal processing applications at NIRA Dynamics since 2007, and target tracking applications at Saab Dynamics in Linköping between 1997-2002. His research interests include positioning and tracking applications mainly using particle filters.
WWW: `www.control.isy.liu.se/~rickard` ,
Mail:`rickard@isy.liu.se`

F. Gustafsson received the M.Sc. degree in Electrical Engineering in 1988 and the Ph.D. degree in Automatic Control in 1992, both from Linköping University, Linköping, Sweden. He is Professor of sensor informatics at the department of Electrical Engineering, Linköping University. His research is focused on sensor fusion and statistical methods in signal processing, with applications to aerospace, automotive, audio and communication systems. He is the author of four books, over hundred international papers and fourteen patents. He is also a co-founder of three spin-off companies in these areas. Prof. Gustafsson is an associate editor of IEEE Transactions on Signal Processing.
WWW: `www.control.isy.liu.se/~fredrik`,
Mail: `fredrik@isy.liu.se`

References

[1] DARPA, “The Grand Challange and Urban Challange,” http://archive.darpa.mil/grandchallenge/.

[2] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: systems and algorithms," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, 2011.

[3] Google, "Google – the self driving car," http://www.google.com/selfdrivingcar.

[4] K. Jo, J. Kim, D. Kim, C. Jang, and M. Sunwoo, "Development of autonomous car part i: Distributed system architecture and development process," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 7131–7140, Dec 2014.

[5] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Tranactions of the American Society of Mechanical Engineering — Journal Basic Engieering*, vol. 82, no. Series D, pp. 35–45, Mar. 1960.

[6] S. F. Schmidt, "Application of state-space methods to navigation problems," *Advances in Control Systems*, vol. 3, pp. 293–340, 1966.

[7] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, pp. 197–208, 2000.

[8] A. Doucet, N. J. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump Markov linear systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 613–624, Mar. 2001.

[9] R. Chen and J. S. Liu, "Mixture Kalman filters," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 62, no. 3, pp. 493–508, 2000.

[10] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings-F Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[11] I. Skog and P. Händel, "In-car positioning and navigation technologies—a survey," *IEEE Transactions on Intelligent Transport Systems*, vol. 10, no. 1, pp. 1454–1469, Mar. 2009.

[12] W. J. Fleming, "Overview of automotive sensors," *IEEE Sensors Journal*, vol. 1, no. 4, pp. 296–308, Dec 2001.

[13] ——, "New automotive sensors — a review," *IEEE Sensors Journal*, vol. 8, no. 11, pp. 1900–1921, Nov 2008.

[14] D. Titterton and J. Weston, *Strapdown Inertial Navigation Technology, 2nd Edition*. IET, 2004.

[15] ESRI, "ESRI shapefile technical description – an ESRI white paper," Jul. 1998.

[16] "Open street map," http://openstreetmap.org.

[17] S. Ammoun, F. Nashashibi, and A. Brageton, "Design of a new GIS for ADAS oriented applications," in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, June 2010, pp. 712–716.

[18] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *Pervasive Computing, IEEE*, vol. 7, no. 4, pp. 12–18, Oct 2008.

[19] C. Whitea, D. Bernstein, and A. Kornhausera, "Some map matching algorithms for personal navigation assistants," *Elsevier Transportation Research Part C: Emerging Technologies*, vol. 8, no. 1-6, pp. 91–108, 2000.

[20] J. Du and M. Barth, "Bayesian probabilistic vehicle lane matching for link-level in-vehicle navigation," in *Intelligent Vehicles Symposium, 2006 IEEE*, Jun. 2006, pp. 522–527.

[21] D. Obradovic, H. Lenz, and M. Schupfner, "Fusion of sensor data in siemens car navigation system," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 1, pp. 43–50, 2007.

[22] M. A. Quddus, W. Y. Ochieng, and R. B. Noland, "Current map-matching algorithms for transport applications: State-of-the art and future research directions," *Transportation Research Part C: Emerging Technologies*, vol. 15, no. 5, pp. 312 – 328, 2007.

[23] O. Noren, "Monitoring of road surface conditions," Master's Thesis LiTH-ISY-EX–14/4773–SE, Dep. of Electrical Engineering. Linköping University, Sweden, 2014.

[24] Y. Cui and S. S. Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 15–25, Feb 2003.

[25] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (SLAM): Part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[26] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 108–117, Sep. 2006.

[27] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent Robotics and Autonomous Agents. Cambridge, MA, USA: The MIT Press, 2005.

[28] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM a factored solution to the simultaneous localization and mapping problem," in *Proceedings of the AAAI National Comference on Artificial Intelligence*, Edmonton, Canada, 2002.

[29] J. Levinson and S. Thrun, "Map-based precision vehicle localization in urban environments," in *Robotics: Science and Systems*, 2007.

[30] N. Mattern, R. Schubert, and G. Wanielik, "High-accurate vehicle localization using digital maps and coherency images," in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, 2010.

[31] Y. Zhao, *Vehicle Location and Navigation Systems*. London: Artech House, 1997.

[32] F. Gustafsson, F. Gunnarson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, Feb. 2002.

[33] F. Gustafsson, U. Orguner, T. B. Schön, P. Skoglar, and R. Karlsson, "Navigation and Tracking of Road-Bound Vehicles," in *Handbook of Intelligent Vehicles*. London: Springer, 2012, pp. 397–434.

[34] M. Ulmke and W. Koch, "Road-map assisted ground moving target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 4,

pp. 1264–1274, Oct. 2006.

[35] D. Salmond, M. Clark, R. Vinter, and S. Godsill, "Ground target modelling, tracking and prediction with road networks," in *International Conference on Information Fusion*, July 2007.

[36] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proceedings of American Control Conference*, Seattle, WA, USA, Jun. 1995, pp. 1628–1632.

[37] K. Jerath and S. Brennan, "GPS-free terrain-based vehicle tracking on road networks," in *American Control Conference (ACC), 2012*, June 2012, pp. 307–311.

[38] S. C. Kramer and H. W. Sorenson, "Recursive Bayesian estimation using pice-wise constant approximations," *Automatica*, vol. 24, no. 6, pp. 789–801, Nov. 1988.

[39] C. Andrieu and A. Doucet, "Particle filter for partially observed Gaussian state space models," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, vol. 64, no. 4, pp. 827–836, 2002.

[40] T. Schön, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear / nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, Jul. 2005.

[41] T. B. Schön, R. Karlsson, and F. Gustafsson, "The marginalized particle filter in practice," in *Proceedings of IEEE Aerospace Conferance*, Big Sky, MT, USA, Mar. 2006.

[42] P. Hall, "A Bayesian approach to map-aided vehicle positioning," Master's Thesis LiTH-ISY-EX-3102, Dep. of Electrical Engineering. Linköping University, Sweden, 2001.

[43] Y. Cheng and T. Singh, "Efficient particle filtering for road-constrained target tracking." *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, pp. 1454 – 1469, Oct. 2007.

[44] N. Svenzén, "Real time map-aided positioning using a Bayesian approach," Master's Thesis LiTH-ISY-EX-3297, Dep. of Electrical Engineering. Linköping University, Sweden, 2003.

[45] J. Kronander, "Robust vehicle positioning: Integration of GPS and motion sensors," Master's Thesis LiTH-ISY-EX-3578, Dep. of Electrical Engineering. Linköping University, Sweden, 2004.

[46] G. Hedlund, "Map aided positioning using an inertial measurement unit," Master's Thesis LiTH-ISY-EX-4196, Dep. of Electrical Engineering. Linköping University, Sweden, 2008.

[47] P. Davidson, J. Collin, J. Raquet, and J. Takala, "Application of particle filters for vehicle positioning using road maps," in *Proceedings of the 23rd International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2010)*, Sep. 2010, pp. 1653–1661.

[48] P. Davidson, J. Collin, and J. Takala, "Application of particle filters to a map-matching algorithm," *Gyroscopy and Navigation*, vol. 2, no. 4, pp. 285–292, 2011.

[49] G. Hendeby and F. Gustafsson, "Sensor Fusion App: sensor fusion at Linköping university." https://play.google.com/store/apps/details?id=com.hiq.sensor.

[50] R. Karlsson and F. Gustafsson, "Velocity estimation (n2515-028-wop00)," *Patent application filed.*, Aug. 2015.