Concept for building a smartphone based indoor localization system

Thomas Willemsen, Friedrich Keller, Harald Sternberg
Geomatik
HafenCity University
Hamburg (Germany)
firstname.lastname@hcu-hamburg.de

Abstract— **Global Navigation Satellite Systems (GNSS)-based navigation with smartphones is very popular. But in areas where no GNSS signal is found navigation could be useful. Examples are navigation in shopping malls, in big offices, in train stations or museums. The goal is to estimate the position in GNSS shaded areas to make navigation possible. The MEMS sensors (Micro Electro Mechanical System) installed in current smartphones, such as accelerometer, gyroscope, magnetic field sensor and barometer allow now navigation also in GNSS shadowed areas. Due to the low quality of these sensors, however, support of the position estimate is needed. In this work, a concept is presented for the construction of an indoor navigation system based on low-cost sensors of smartphones. The position estimate from the available sensor data forms the basis of the position determination. So position estimation is always possible independent of location. First results with Kalman filter and particle filter are shown. The presented concept serves as a basis for the construction of a smartphone-based navigation solution for indoor use. Therefore the available MEMS sensors should be used as a position estimator and a wide variety of supporting information can be processed. A first approach for implementation on a smartphone is shown as an example.**

Keywords: indoor navigation; MEMS; Kalman filter; particle filter; smartphone

I. Introduction

GNSS based navigation systems has become an essential component of everyday life. In areas where no GNSS signals can be received this system is failing however. But even in these areas route guidance may be useful. Examples of this could be the navigation at exhibition grounds, in shopping malls, at train stations or in large office buildings. Within the research field of indoor navigation they work hard on technologies which enable an indoor positioning in GNSS shadowed areas.

Previous technologies for indoor positioning determine distances to reference points wirelessly. They use technologies, such as Radio Frequency Identification (RFID), wireless, ultrasonic, optical methods and also the magnetic field [1]. There are already dedicated solutions (selection [1]: awiloc Fraunhofer, Radianse, Ubisense, LPR-2D Symeo, Active Bat-AT & T), which are often implemented at great expense. Due to high demands on the infrastructure and the clients implementations can often be realized in a single room / building or a factory only. To minimize the effort of the infrastructure measures inertial sensor is increasingly used, which is for example also implemented as MEMS design (Micro Electro Mechanical System) technology in smartphones.

Due to the efficiency and the quantities of these sensors in a smartphone, a navigation solution with this sensor is obvious. The heavily widespread use of smartphones in society, for example by the GNSS-based navigation is another decision criterion. The goal is to develop an algorithm which is based solely on smartphone sensors. Therefore only minor interventions are necessary in the infrastructure of the building intended for navigation. The accuracy of the position estimate should be here between 1-5 m with the condition of a room-exact position.

Fig. 1 gives an overview of the concept to implement a smartphone-based indoor navigation system. This paper is also structured in accordance with this concept. After selecting the navigation hardware under consideration of the application and the user group to be addressed, the study of the quality of the integrated low-cost sensors can be done. This will be treated in the following section using the example of the barometer for floor detection and calibration of the accelerometer. Then an explanation of the implementation of the position estimate is made. The fusion of sensor data using the Kalman filter and particle filter is presented as an example.

The position estimation with low inertial sensors is only for a short time accurate and must therefore revert to external support. Possibilities of support can be versatile. In this work the fingerprinting and the position determination is presented by means of trilateration.

In addition to the position estimate, route guidance is also necessary for a navigation system. Various routing algorithms which can make a cost calculation for all possible routes can be used.

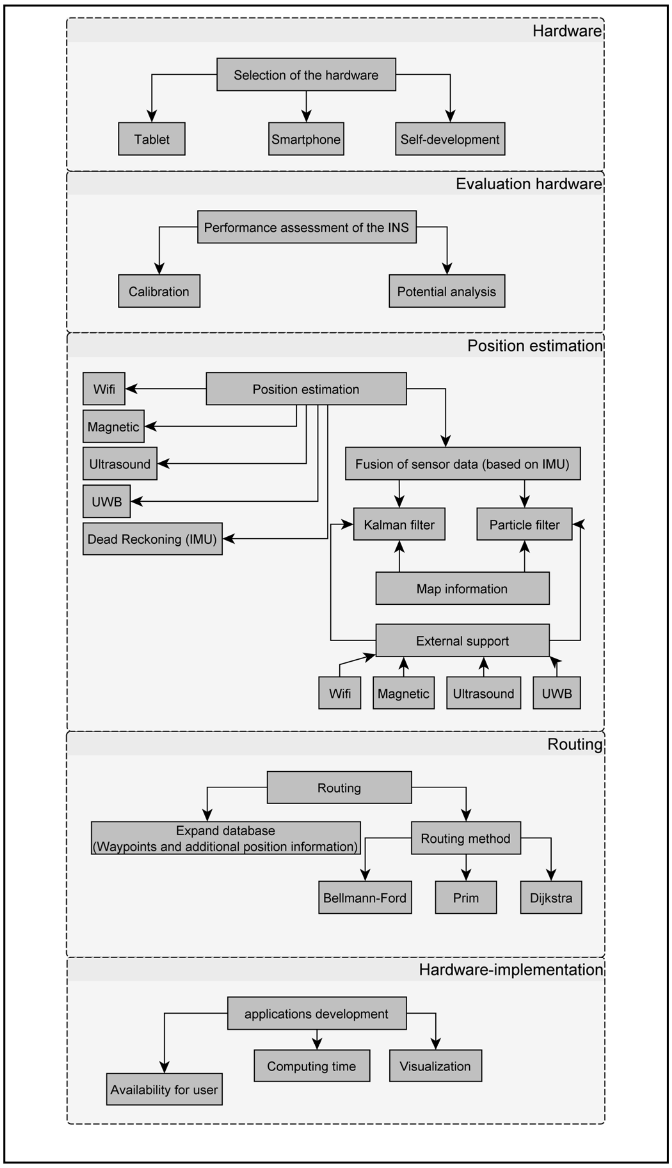

Fig. 1. Concept for building an indoor navigation system.

Finally, an application is built on the basis of all information. Particularly, the computing power of the hardware must be considered, but the visualization and guidance are also of vital importance.

As test devices, the LG E960 (Google Nexus 4) and the Samsung Galaxy Nexus are used in this paper. In size and features the two smartphones make hardly any difference. In the Nexus 4, however, is a newer generation of sensors.

For the following studies presented by means of a separate application on the basis of Android [2], the sensor data, such as the accelerometer, the gyroscope, barometer and the magnetic field sensor are recorded on the smartphone. The algorithms will be developed afterwards with Matlab. After testing the post-processing successfully the algorithms will be ported in an application on the smartphone to test also realistic, real-time applications.

II. Evaluation of Smartphone Integrated MEMS

The investigation of the position estimate of the available sensors forms the basis of fusion of the sensor data. A usability of air pressure to the floor detection, as well as the calibration of the accelerometer are presented according to [3] and [4].

A. Barometer

The BMP180 from Bosch which is installed in the experimental unit can be used in the indoor navigation as floor detection. To this end, the detected air pressure p_i is converted with the barometric height formula (1) into relative height h_i. For the barometric formula, the sea level with a local middle atmosphere is used as a reference.

$$h_i = \left(1 - \sqrt[5.255]{\frac{\mathrm{p}_i}{1013.25}}\right) * \frac{288.15}{0.0065} \qquad (1)$$

This test should identify if and in which resolution the floors can be distinguished. For floor recognition, a staircase in Building D of the HafenCity University (HCU) was used. The measurement range was over all four floors. As part of an annual survey exercises with students the storey heights are known cm-precisely. While the testing device register continuously measured data through its own application, the floors were walked along. On each floor the device was positioned 4 minutes. The floors were visited with the sequence 4-3-2-1-1-2-3-4. The result of Nexus 4 shows Fig 2.

The blue lines represent the reference heights. In black the raw data air pressure is converted directly in relative heights and green is a smoothing to 1 second. The reference level was the average of the first floor. The non-smoothed determined heights rush in the range of ± 1 m. The results show clearly that floors can be distinguished only by means of the barometer.

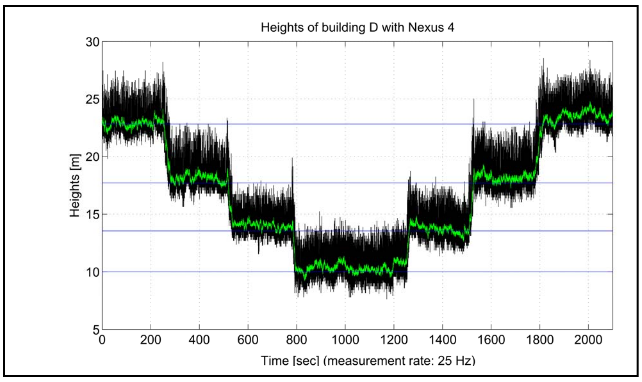

Fig. 2. Floor detection with barometer data.

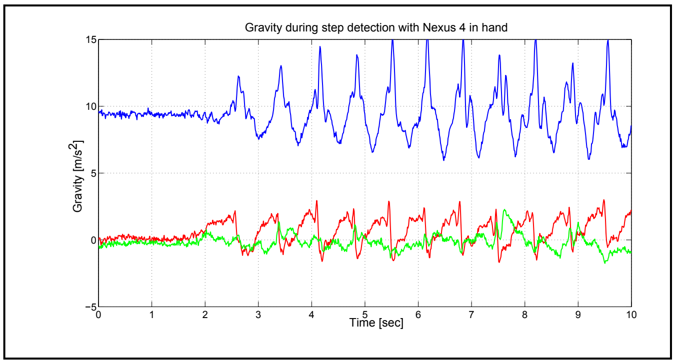

Fig. 3. Smartphone accelerometer data during walking.

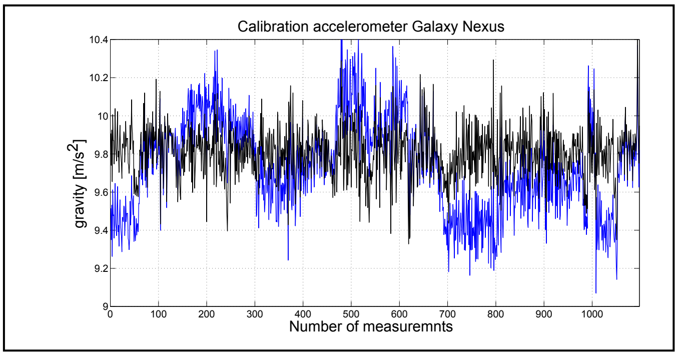

Fig. 4. Magnitude accelerometer data Galaxy Nexus (raw data = blue; calibrated = black).

B. Accelerometer

The magnitude of the 3-axis accelerometer shows gravity on standstill. This can be used to level the smartphone coordinate system during the navigation. From a spatial position estimation a separation in position and height can be performed then.

The double integration of the acceleration leads to the travelled distance. Due to the strong drift of the MEMS accelerometer the system provides unusable results. Therefore, the accelerometer is used as a pedometer. Fig. 3 shows data of the 3-axis accelerometer from the testing device during a brief stoppage and a maturity period. Here, the phone was held freely in the hand before the body.

A calibration for use as a pedometer does not seem to be necessary at the moment. But since a leveling of all sensors in the smartphone should be realized via the accelerometer, false accelerations can produce an erroneous reference of the situation. The result would present rotation rates, which are integrated to angles of rotation in the wrong system.

For the calibration of the accelerometer, use is made of the local target values of the gravitational acceleration. These can be obtained for Germany from the Physikalisch-Technische Bundesanstalt (PTB). The data used here are from 2007 [5]. The calculation methods favored [6] a Kalman filter. According to this technique, the smartphone accelerometer is calibrated. As a functional context of the calibration the spatial Pythagoras is used with (1):

$$g^2 = a_{xref}^2 + a_{yref}^2 + a_{zref}^2 \qquad (1)$$

With the introduction of $g = 9.813\ m/s^2$ as a reference for Hamburg and the relationship between the reference and the measured value with the correction variables bias and scale (2-4):

$$a_{xref} = s_x\ (a_x + b_x) \qquad (2)$$

$$a_{yref} = s_y\ (a_y + b_y) \qquad (3)$$

$$a_{zref} = s_z\ (a_z + b_z) \qquad (4)$$

The functional model can be set up and the correction parameters scale s and bias b of each axis can be compensated.

A detailed structure of the Kalman filter to determine the correction parameters of the 3-axis accelerometer can be found in [4].

To determine sufficient and significant correction parameters the phone needs to be turned several times on its own axis during data production. The goal is to merge the individual axes of the accelerometer positively and negatively on the acceleration of gravity. For optimal measurement data it is necessary to rotate around the sensor center as good as possible and to maintain a stable rotational behavior, so that no additional accelerations will emerge during calibration.

Fig. 4 shows an example of a Galaxy Nexus data record in blue the raw data as magnitude of the three axes of the accelerometer, as well as in black, the corrected magnitude with the scale and bias determined by Kalman filter. An improvement of the sensor data by the calibration can be seen. If the non-calibrated data is used, like in this example, a wrong leveling about 1° is generated. So a part of the real rotation rate around the z-axis will be detected on the other axis (x,y). The effect during position estimation is an additional drift in the orientation.

III. Fusion Kalman filter and Particle filter

The calculation of a position by inertial sensors can be implemented by Dead Reckoning. For the purpose of persons navigation the rotation rates are often integrated and the accelerometer is used as step detection. With the knowledge of the step length and the present orientation during step detection a relative position determination can be carried out with polar appending.

Since the position estimation of gyroscope and accelerometer data is not sufficient and provides only for a short time enough accurate position information, the position estimation needs support. The use of filter is the easiest way to unite supporting information with sensor information. This simplifies the merging of different stochastic information. Hereafter examples of Kalman filter and particle filter will be presented.

In both experiments, a support by maps is implemented. The map is used to visualize and is considered in this concept as present. The map format is a simplified xml format in which each line of the wall is defined by two coordinate pairs. The so-prepared filter approaches are feasible on the smartphone regardless of infrastructure and form the basis for the consideration of external support.

A. Kalman filter

In a first step, the sensor data are prepared. These will be registered in a smartphone partly up to 200 Hz. In order to minimize the noise, the sensor data (not gyroscope) will be averaged before use in the Kalman filter. The height information in the Kalman filter is determined by the relative heights of air pressures of the barometer. For this purpose, the air pressure is calculated directly to a relative height by the barometric formula. The relative heights will also provide later for the staircase detection in pedometer use, so that the step length can be customized.

From the data of the accelerometer, the inclinations for the x- and y-axis of the smartphones are determined in order to calculate the leveling for all 3-axis sensors. The actual 3-D position determination is thus a 2D + 1D system. The height of the barometer can be calculated independently of the position. This can be an independent support by the height. A change in altitude can serve as a support position to identify stairs or elevators in the area. The magnetic field sensor and the gyro are leveled and the orientation can be determined in the plane for both sensors. This allows the support between the rotation r_z from the gyroscope to the computed magnetic north if the gyroscope is initialized with the azimuth of magnetic north.

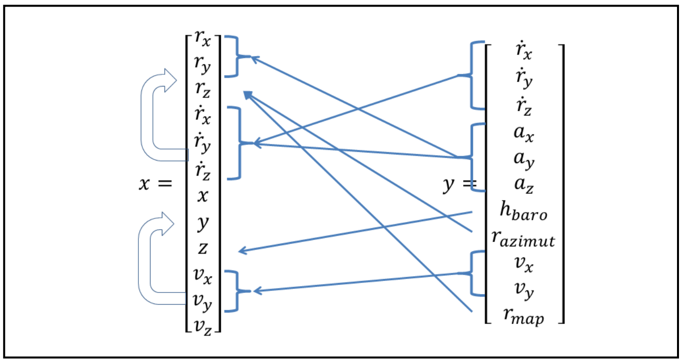

Fig. 5. Relationship between state vector and measurement vector.

The detection step of the accelerometer which is providing the current speed v_x, v_y in this filter operates initially with a limit value for a positive acceleration along the z-axis, and the search for a negative slope afterwards. The step length is initially set to be known. Every step-recognition is followed by support of speed depending on r_z in the Kalman filter. In case the known step length is extremely wrong, large inaccuracies in position determination can occur. Previous works on step length determination try to estimate the step length [7, 8, 9] by step frequency, amplitude of accelerometer signal and striking characteristics such as size and sex. The step lengths can be determined with an error of about 10%. The step length determination or stride support in the filter should be the focus on investigations afterwards.

The connection between the state vector and measurement vector is presented in (Fig. 5). The motion model, shown here as broad arrows, provides an update of the horizontal orientation and the 2D +1 D coordinates by the corresponding speeds.

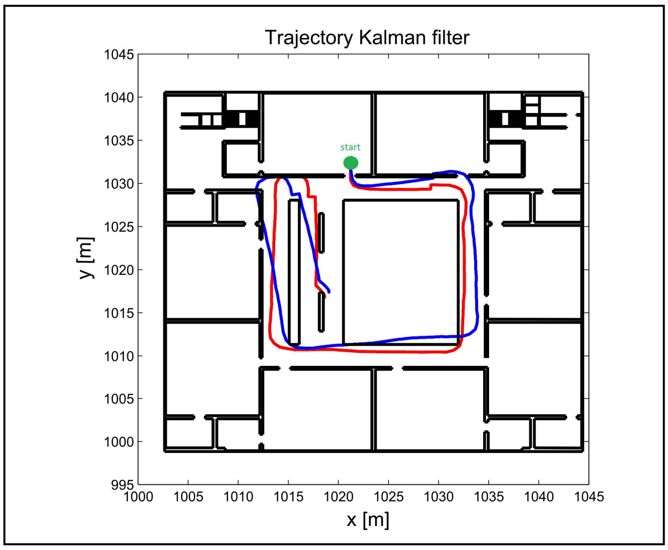

Fig. 6. Trajectory with Kalman filter (blue = only dead reckoning; red = KF in addition with map information.

The rotation rates and accelerations in the measurement vector are used to support the leveled rotation rate in the state vector. Accelerations and g support the inclination angle r_x and r_y. The height is corrected directly by the relative height of the barometer and the position coordinates (x, y) by means of speed from the pedometer. To minimize the drift out of the integrated rotation rate, a support of r_z is additionally provided via r_{azimut} from magnetic sensor data and r_{map} from the direction information of the adjacent wall. The Kalman filter is initialized with the current orientation to the north and the current 3D position (e.g. from a QR code) to enable support from magnetic sensor data is directly possible. Before the filter starts, an alignment is carried out over the period of 6 seconds to determine the current offset of the rotation rate. This offset produces the drift through integration of the rotation rate.

Fig. 6 displays the first results in a building of HafenCity University. The first section of a trajectory is shown, which starts in the office upper left and continuous to the stairway where the floor switch takes place. In blue, the Kalman filter without the support of the directional information of adjacent wall lines can be seen. In red, the support of the wall is taken into account. It is clearly a correction by map support.

For the implementation of the map support, the straight section was used. Current position and direction of the person navigating define the first straight line and the respective vertices of the adjacent wall define the second straight line. The adjacent wall is selected under the condition of the minimum orthogonal spacing. So that the direction support takes place, the foot of the current position must be on the wall. This ensures that in curves will be little to no support. If the conditions are met, the direction of the wall is transmitted into the measurement vector and used as support of r_z.

B. Particle filter

The sensor data are also prepared for the particle filter, as already in the Kalman filter. The position estimation of the particle filter is implemented using distributed particles and their proper balance to the next likely condition. The use of known patterns in the distribution of these particles allows the reduction of the necessary number of particles. This helps to keep the computational effort low. Additional informations about the next possible position are thus very helpful. The structure of a particle filter is described in [4]. In line with this, the particle filter was implemented for this position estimate.

After a normally distributed initialization of the particles, depending on the walking speed, the particles are spread in the ongoing process over the current direction and the walking speed. In this case, the direction and the walking speed are noisy in the expected error behavior. Support in the step detection is carried out by means of the probability density function with coordinate of Dead Reckoning to the respective particles.

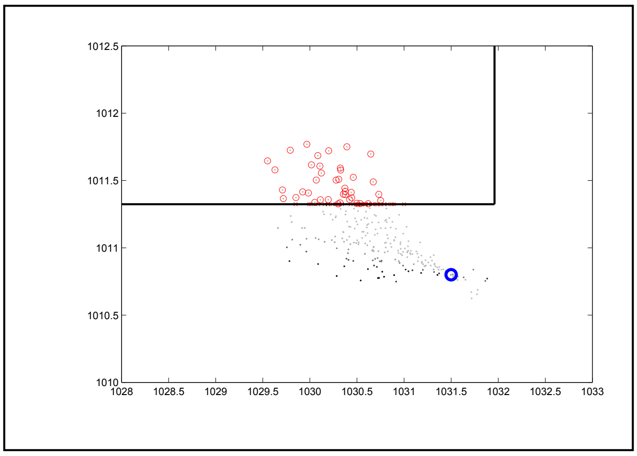

Fig. 7. Step by particle filter: principle of supply with map data. (Blue = last actual position, red = deleted particle, grey = lower weighted particle, black = higher weighted particle)

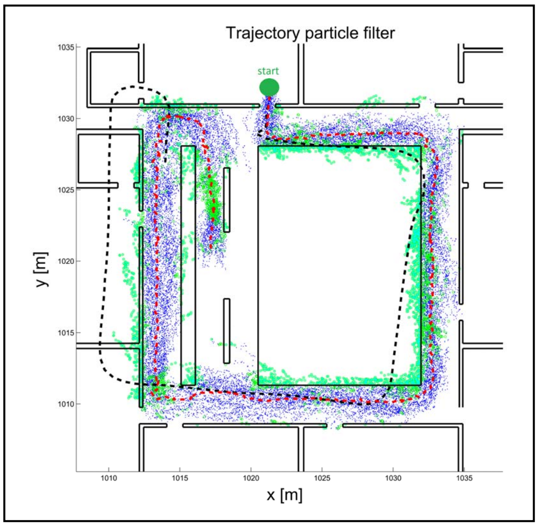

Fig. 8. Particle filter with supply by map data.

As a first map support, a calculation of section from the adjacent walls is carried out in every step for every particle. The last and most likely position and the respective particle define therefore the straight line. When the intersection is located on the wall or between the last most likely position and the current particles, then the weight of the particle is set to 0 (Fig.7).

A second support with the adjacent wall is performed by the directional comparison. For all other particles, the direction to the latest most likely position is determined and related to the wall direction by means of a probability density function. Fig. 7 shows for a step, the effects of weights through this support. Blue represents the last most likely position. Red particles were turned off by the weight of 0. The value range of the still current particles is from black for high weights up to light gray for low probability position.

Fig. 8 shows with the same data set of the Kalman filter, a first implementation. The red trajectory represents the most likely positions. Blue are the particles and black is the Pedestrian Dead Reckoning. It can be clearly seen how the wall geometries support the trajectory. Particles behind walls are excluded (cyan) and by means of directional information of the nearest wall particles are upgraded.

The examples give an overview of the merger of existing, infrastructure independent information. The processing of additional supporting information in these filters is presented as follows in principle.

IV. SUPPLY WITH ADDITIONAL DATA

As external support or also as a standalone solution for position estimation, different technologies are used. Examples would be WiFi, magnetic field, Ultra Wide Band, Ultra Sound and optical systems too. Optical systems often used for direction support through image sequences and the other known technologies can be divided into two methods. The fingerprinting and support by trilateration.

A. Fingerprinting

This method for position estimation is usually used in conjunction with WiFi access points. This requires a large number of routers to be distributed in the building which is to navigate. The position is estimated via comparison of currently registered signal strengths receivable router to a reference database. The reference database contains beside the position information also the associated signal strengths of the receivable access points. To produce this reference database, the building needs to be walked on the known way points after distributing the access points (Fig. 9 right). On every way point, the WiFi is received and recorded. In the reference database, these data are assigned to the corresponding position. During navigation, the WiFi is received through the navigation hardware. In the position computation these data (Fig. 9 left) will now be compared with the database through a minimization function. Interpolations also allow position determination between the reference points.

For the implementation of fingerprinting a relatively large number of routers must be stationed in the building. In addition, directional antennas could be partly used for sufficiently good position accuracy. The implementation effort is therefore correspondingly high, if it is used as the only position estimate. By combining with inertial sensors, the costs of infrastructure measurements can be minimized since no comprehensive position determination by means of fingerprinting is required. Thus, even fingerprinting is suitable as supporter of an IMU-based position estimate.

With the system *awiloc*, the Fraunhofer Institute implemented an indoor positioning service based on fingerprinting. The position estimation is performed exclusively by wireless fingerprinting. So far, 60 projects were implemented for different applications with the *awiloc* system [10].

B. *Trilateration*

The wireless signal strengths (WiFi, Bluetooth, UltraWideBand) can be converted to distances by means of signal attenuation function. In ultrasound systems, the term of signal is measured and converted with the known speed of sound into a distance. UWB and ultrasound cannot currently be received in the test devices. Therefore an additional external receiver must be connected to the smartphone. The data could be sent via Bluetooth to the application in the phone.

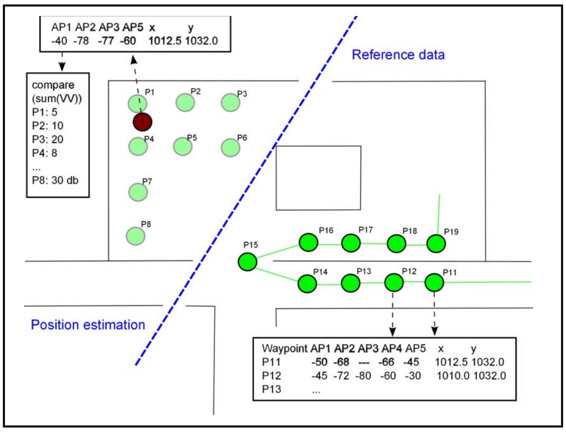

Fig. 9. Principle of WiFi fingerprinting in indoor positioning. Right: generating reference data; Left: principle of position estimation with reference data.

If those systems are directly used as position sensor, there must be at least two routes or three lines in 3D space to reference points (access points) be available to determine a clear position by 2D or 3D curved section (Fig. 10). For better reliability, more routes should be available in order to exclude possible rough measurement uncertainties of the calculation. A robust estimator can be found in [11]. The functional context is the 2D or 3D Pythagoras with:

$$S_i^2 = (x_{APi} - x_P)^2 + (y_{APi} - y_P)^2 + (z_{APi} - z_P)^2 \qquad (1)$$

The Pythagoras illustrates the relationship between the measured distances S_i, the corresponding known access point coordinates x_{APi}, y_{APi}, z_{APi} and the wanted position x_P, y_P, z_P.

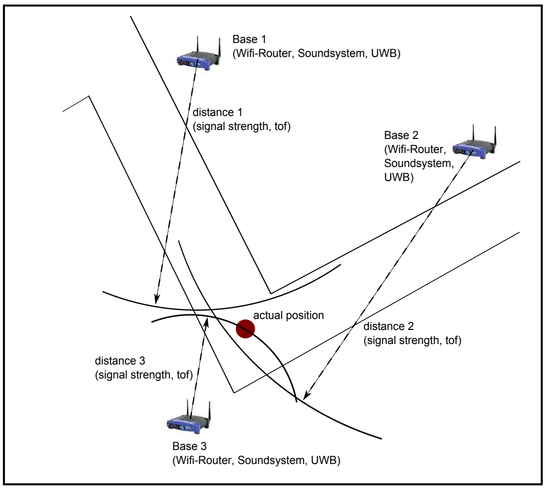

Fig. 10. Principle of Trilateration in indoor positioning.

Costs can also be minimized here, by combining inertial sensors with the infrastructure. Availability throughout the building is not necessary, since gaps can be bridged by Inertial Navigation System (INS). In addition, a support is not necessarily made by a previously calculated position of several distances but individual route information can be used directly for support. In the particle filter, this can be relatively easily implemented, since a particle gains a higher weight when the distance to the access point is approximately kept. The Kalman filter requires for support with only one distance, additional information about the direction to the current position and the access points.

After implementation of a position estimate by using the previously presented options, the algorithms must be implemented in an application. Field tests then serve the examination of the algorithms also on the navigation device.

V. EVALUATION ON SMARTPHONE

Current smartphones must meet demanding system specifications. The processor and the RAM must keep pace with the previous computer generation. For this reason, the applications for smartphones have grown steadily, and previous applications are converted with considerably more effort. The need to minimize the computation time for a navigation application is nevertheless unavoidable. The update rate of the sensors is a primary factor. Calculation sequences in an application that require more time reduce the number of possible sensor readings. This can lead to a faulty position estimate, and fewer steps are detected. Raster data, such as vector data, are considered as a map basis. However, considerable computing time is required for the visualization of the map data. Techniques from augmented reality can be applied. A simple arrow showing the direction of movement can be projected in the display, for example. Using existing raster data for navigation, the calculations can be significantly reduced. Fig. 11 shows the application (App) in which the approach described previously for the fusion of the smartphone sensors using a Kalman filter is implemented in the test smartphone.

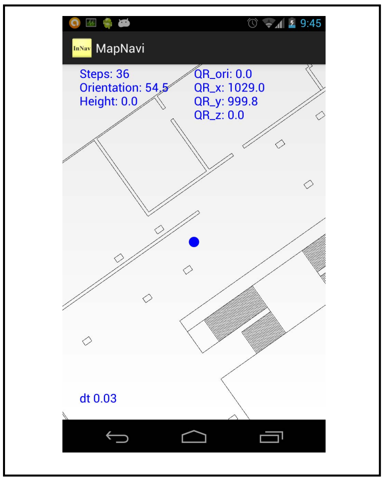

Fig. 11. First approach for a navigation application (App) in the test device.

The Kalman filter is initialized using a QR-code. The smartphone display shows the map of the first floor of Building D at HCU in vector data format. The circle in the center of the smartphone display indicates the current position. During navigation, the map moves to maximize the view around the current position on the map. In the upper area of the display, information such as the current orientation, the current height and the starting position and orientation, which may be from a photographed QRcode, are displayed. In the lower left corner, the time difference dt (in seconds) between two calculation runs is indicated. For vector data, this value averages 0.03 seconds when running on the Nexus 4 test device. If the same application is used with the associated raster map, the computing time increases in the middle of a run to 0.3 seconds. In this time period sensor data not get stored. The peaks in the accelerometer data will not be detected and cannot be used for step detection. Consequently, not every step will be detected with the current approach to step detection with the raster map.

Furthermore, the choice of the map format has a significant effect on calculations that support navigation such collision detection (e.g., with walls). The detection can be performed with raster data using occupancy maps or with vector data using intersection calculations. Moreover, a classification of the map elements is of interest. Lines must be differentiable for stairs, walls and doors, for example. With vector data, a classification can be included in the xml format, for example. With raster data, a color variation of the respective class would be necessary. These algorithms can be refined for use in map matching. The user can move only on classified edges and nodes that were defined beforehand in the building model. The visualization and the calculation model can be implemented in two dimensions or three dimensions for smartphone-based navigation. In the current case of a 2D+1D solution, a two-dimensional solution is more suitable for display. In an extension through external support, for example through WiFi or a similar measuring device, a three-dimensional model may be better-suited for correction purposes. An additional aspect is how people interact with the map data. A study was conducted to decide if it is sufficient to guide people to their destination by simple arrows or if an enhancement with maps is beneficial. It was found that with maps, users found their way much faster, especially when lost; a detailed discussion can be found in [12].

The current application using the Kalman filter for the fusion of the smartphone sensor data is a basis for further study on the optimal base map for indoor navigation with smartphones.

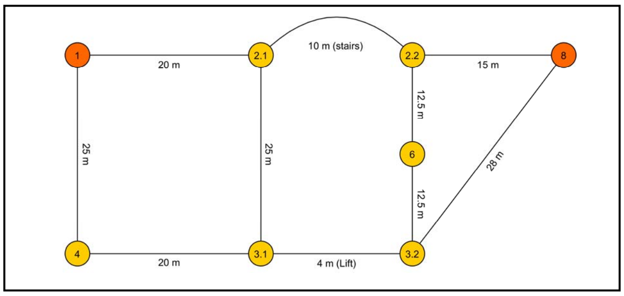

Fig. 12. Principle for way points in maps to do cost calculation for routing.

VI. Routing on Smartphone

To complete the navigation application, a route guidance must be implemented. There are a number of routing algorithms. Using the example of Dijkstra algorithm the implementation should be described here. The algorithm can be classified into the group of greedy algorithms.

The basis of the routing algorithms forms is graph theory. A graph is defined by n-nodes which are connected by edges. Fields of application for graphs are communications networks, way planning, search engines and object-oriented programming languages. For the putting into action of routings undirected and directed graphs can find use. The edges have a confirmed direction of movement at undirected graphs. However, a mixture can also be converted in the routing [13]. The edges get weighted in the routing. Costs can be the time expenditure and also the distance. The costs describe the effort for the partial way.

The implementation of routings is less complex in the building opposite than in street navigation. There is only speed. The costs therefore can be determined directly by the infrastructure, for example distance, stairs and lift.

TABLE I. Routing Cost Calculation

Different ways between 1 and 8 in Fig. 10	
Points	***Costs[m]***
1 – 2.1 – 2.2 – 8	45
1 – 2.1 – 3.1 – 3.2 – 8	77
1 – 4 – 3.1 – 3.2 – 8	77
1 – 2.1 – 2.2 – 6 – 3.2 – 8	83
1 – 2.1 – 3.1 – 3.2 – 6 – 2.2 – 8	89
1 – 4 – 3.1 – 3.2 – 6 – 2.2 – 8	89
1 – 4 – 3.1 – 2.1 – 2.2 – 8	95

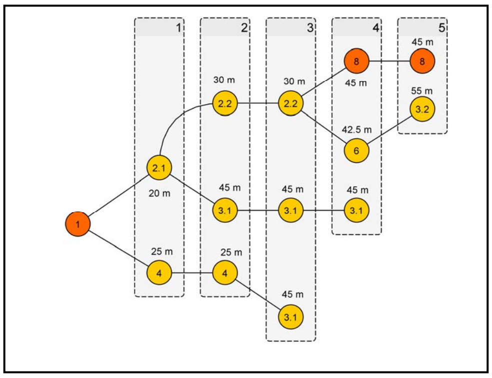

Fig. 13. Tree of calculation best route.

An example of non-directional way point net is shown in Fig. 12. To combine the different floors, a connection with stairs or lift is necessary; see Fig. 12 way points 2.1 to 2.2. The shortest way between 1 and 8 is looked for. The possibilities are shown in TABLE I. In this example the shortest way is chosen 1-2.1-2.2-8. In the approach from Dijkstra the smallest costs from neighboring points are continued in the next step. Fig. 13 shows the route finding. The calculation ends, if the arrival point is found and all other routes have the same or more costs. The computing time depends on the number of way points and edges.

CONCLUSION

Due to the wide spread of smartphones and the strong use of GNSS-based position-imaging applications, the extension to navigation in GNSS shadowed areas is a very interesting application.

This work is intended to show a possible workflow for the construction of an indoor navigation system. To this end, the smartphone is the preferred navigation hardware. Using the example of the barometer and the accelerometer an evaluation of the smartphone sensors for indoor positioning was presented. The air pressure of the barometer provides height informations, which make it possible to detect the change of floors during navigation.

The comparison of both smartphones shows that a calibration of the Galaxy Nexus seems to be necessary, while the sensors in the Nexus 4 use a system calibration. After calibration, both smartphones produce the same quality of trajectories.

A combination of sensors with the support of map information in vector format creates an autonomous position determination and thus is independent of buildings. In each case, an approach with Kalman filter and particle filter was presented. It could be shown that a position estimate with both methods is possible. But the processing of supporting information represents a significant difference. Future work will therefore focus on the support of external information comparatively in the Kalman filter and particle filter.

The implementation of an application on the smartphone is mainly depending on the computing power of the system and the possibilities for visualization. If the way point network for route guidance is available during the navigation, it is possible to use it as support. Then the complex support by map data is unnecessary.

As the examples of the respective concept show, the effort to set up an indoor navigation system is largely defined by the implementation of the position determination. A combination with inertial sensor minimizes the effort of cost intensive additional installations in the building. Next steps are the evaluation of the developed fusion algorithms with additional support data as well as tests in support-critical areas. A routing algorithm will be implemented on the smartphone and a support by means of way point network will be investigated, which could replace the map data support.

REFERENCES

[1] Mautz, R. (2009): „Indoor Positioning – Präzise Verfahren zur Innenraumpositionierung“, (*Indoor Positioning - precise methods for Indoor Positioning*), geodätisches Kolloquium der TU Darmstadt 2009/2010 http://www.igp.ethz.ch/GSEG/people/rmautz/mautz_darmstadt09.pdf Retrieved: 01. November 2012

[2] Android developer (2013): http://developer.android.com/index.html Retrieved: 03. Juli 2013

[3] Keller, F., Sternberg H., Willemsen T. (2012): “Calibration of Smartphones for the use in indoor navigation”, International Conference on Indoor Positioning and Indoor Navigation (IPIN), 13-15 November 2012, Sydney, Australia.

[4] Willemsen T., Keller, F., Sternberg H., (2013): “Kalibrierung von im Smartphone verbauten Sensoren”, (*Calibration of sensors installed in smartphones*), Oldenburger 3D-Tage, 13-14 Februar 2013, Oldenburg

[5] PTB (2007): https://www.ptb.de/de/org/1/11/115/doc/tabelle.pdf Retrieved: 05. Februar 2013

[6] Wendel, J. (2011): “Integrierte Navigationssysteme”, (*Integrated Navigation Systems*) 2. Auflage, Oldenbourg, München, 2011

[7] Kim, J.W., Jang , H.J., Hwang, D.H., and Park, C. (2004): “A Step, Stride and Heading Determination for the Pedestrian Navigation System”, In: Journal of Global Positioning Systems (2004) Vol. 3, No. 1-2: 273-279

[8] Goyal, P., Ribeiro, V., Saran, H., and Kumar, A., (2011): “Strap-Down Pedestrian Dead-Reckoning System” In: International Conference on Indoor Positioning and Indoor Navigation (IPIN), 21-23 September 2011, Guimara (Portugal)

[9] Gädeke, T., Schmid, J., Stork, W., Müller-Glaser, K. (2011): „Pedestrian Dead Reckoning for Person Localization in a Wireless Sensor Network” In: International conference on indoor positioning and indoor navigation (IPIN), 21-23 September 2011, Guimara (Portugal)

[10] Awiloc: http://www.iis.fraunhofer.de/de/bf/ln/technologie/rssi/prj1.html Retrieved: 25. Oktober 2013

[11] Blankenbach, J., Willer, V.: „Robuster räumlicher Bogenschnitt – Ein Ansatz zur robusten Positionsberechnung in Indoor-Szenarien“, (*Robust spatial Resection - An approach to robust position calculation in indoor scenarios*), AVN, Heft 8-9/09, AVN – 8-9/2009

[12] Weninger, B., Bouchain, J., Lukianto, C. and Sternberg, H.: “Smartphone-Based Portable Pedestrian Indoor Navigation: Experiments on Visualisation and Positioning.” In: Gartner, G. und Ortag, F. (eds.) Digital Proceedings of the 8th International Symposium on Location-Based Services (LBS), 21.-23. November 2011, Wien, 2011

[13] Lederer, T. (2009): “Entwicklung eines Navigationssystems – Optimierung von Routing und Visualisierung”, (*Development of a navigation system - optimization of routing and visualization*), VDM Verlag Dr. Müller, Saarbrücken 2009