



Smartphone Navigation anhand von „intelligenten Check-points“

Hannes Hofer und Günther Retscher, Wien

Dieser Beitrag wurde als „reviewed paper“ angenommen.

Kurzfassung

Mithilfe von kompakten GPS Empfängern, die wir in unseren Smartphones finden, ist die Navigation von A nach B sehr einfach geworden. Im urbanen Raum und in Gebäuden ist die Positionierung jedoch oft nicht ausreichend genau oder gar nicht möglich, da die Positionsbestimmung per GPS ungenauer wird, wenn das Signal gestreut, reflektiert oder abgeschwächt wird. Neben GPS verfügen Smartphones auch über andere Bauteile, wie z. B. Bewegungssensoren, Magnetfeldsensor und WLAN-Modul, die zur Ortsbestimmung herangezogen werden können. Das Ziel unserer Forschung war es, die Daten dieser Komponenten zu kombinieren und dadurch die Genauigkeit der Positionierung sowohl im Outdoor- als auch im Indoor-Bereich zu verbessern. Die Grundidee des neuen Ansatzes ist, entscheidende Wegpunkte entlang der Trajektorie intelligent zu wählen und das Passieren dieser mithilfe des WLAN-Moduls zu erkennen und Positionsveränderungen dazwischen mithilfe des Bewegungssensors und des digitalen Kompasses zu ermitteln. Bei unseren Untersuchungen konnten durch die Kombination der Sensoren und Module die Positionen in den Testläufen in kombinierten Indoor- und Outdoor-Gebieten im Mittel auf 2,0 m genau bestimmt werden, während die Abweichung der mittels GPS ermittelten Koordinaten bei 16,7 m lag, sofern GPS überhaupt verfügbar war.

Schlüsselwörter: Navigation mit intelligenten Wegpunkten, WLAN Fingerprinting, Trägheitsnavigation, Schritt-erkennung, Bestimmung der Fortbewegungsrichtung

Abstract

Navigation from a start point A to a destination B has become an easy task due to the integration of compact GPS receivers in our smartphones. Nevertheless, in urban areas and in buildings GPS positions are often not accurate enough or even cannot be obtained at all because of the fact that GPS signals are attenuated, reflected or weakened. Besides GPS, smartphones have other in-built sensors, such as inertial sensors (accelerometers and gyroscopes), and modules for data communication, such as Wi-Fi, which can additionally be used for positioning. The aim of our research is to combine the data of these components and to improve thereby the accuracy of localization in outdoor as well as indoor environments. The basic idea is to choose waypoints along the trajectory intelligently while navigating and to determine position changes in between with the help of the inertial sensors and the digital compass. In our investigation and tests we could achieve positioning accuracy of around 2.0 m on average for combined localization, while the deviations of the coordinates obtained from GPS laid within a range of 16.7 m only if GPS was available at all.

Keywords: Inertial Navigation, intelligent Check Points, Wi-Fi positioning system, step detection, heading determination

1. Technische Grundlagen

Mithilfe des WLAN-Moduls im Smartphone oder Tablet und einem vorhandenen Netz von WLAN-Access-Points (APs) kann auf eine Position geschlossen werden. Eine weitverbreitete Methode ist das WLAN-Fingerprinting, dabei werden die Signalstärken bzw. RSS-Werte (Received Signal Strength) von mehreren APs gemessen und einer Position zugeordnet. Diese Kombination von RSS-Werten für eine Position wird in einer Fingerprinting-Datenbank (FPDB) bzw. Signalstärkenkarte (sog. Radio Map) gespeichert. Diese Phase der RSS Messungen und Zuweisung wird

in der Regel als Kalibrierungsphase oder Trainingsphase bezeichnet.

Des Weiteren können die Daten des Beschleunigungssensors bzw. Accelerometers, des Magnetfeldsensors sowie auch des Barometers im Smartphone ausgewertet werden, um Veränderungen in der Position zu ermitteln. In [1] wird ein Ansatz beschrieben, der den Beschleunigungssensor und den digitalen Kompass benutzt, um Schritte und deren Richtung zu erkennen. Es ist auch möglich, mittels Integration der Beschleunigungssensordaten auf die zurückgelegte Strecke zu schließen [2].

1.1 WLAN-Fingerprinting Nearest Neighbour Algorithmus

In [3] wird ein „Nearest Neighbour“ (NN) Fingerprinting Ansatz, also eine Strategie, bei der der nächste Nachbar in den gemessenen Signalstärkewerten gewählt wird, vorgestellt. Bei diesem deterministischen Algorithmus werden die WLAN-Scans in mit bis zu vier Smartphoneorientierungen mit der dazugehörigen Position in einer FPDB gespeichert. Abbildung 1 zeigt das Erfassen der RSS-Werte für eine Position in einer Orientierung.

Bei der Positionierung eines Nutzers in der sog. Positionierungsphase kann an einer beliebigen Stelle ein WLAN-Scan durchgeführt werden. Als Position wird jene gewählt, deren gemessene RSS-Werte jener in der FPDB am nächsten gelegen sind. Quantifiziert wird dies anhand des euklidischen Abstandes zwischen den RSS-Werten der einzelnen WLAN-Scans. Die Scans für eine Position j können als Vektoren dargestellt werden, welche die RSS-Werte der einzelnen AP_i von 1 bis n enthalten, wobei n die Anzahl der Access Points in der Datenbank darstellt.

Die Berechnung des euklidischen Abstandes d_j zu einer Position j der FPDB ist in Gleichung 1 dargestellt. Darin entspricht $RSS_{m_{AP_i}}$ der zur Positionierung gemessenen Signalstärke für einen AP_i (Index m steht für measured $RSS_{m_{AP_i}}$). Der Wert $RSS_{j_{AP_i}}$ beschreibt den erfassten RSS-Wert des AP_i an der Position j . Beim NN-Algorithmus werden von allen in der FPDB gespeicherten Positionen die euklidischen Abstände zu deren RSS-Werten berechnet. Gewählt wird die Position, deren RSS-Werte den kleinsten euklidischen Abstand d_{\min} zu den gemessenen RSS-Werten aufweist. In diesem Ansatz kann somit einem WLAN-Scan nur eine erfasste Position zugeordnet werden.



Abb. 1: Position der Smartphones bei der Datenerfassung

$$d_j = \sqrt{\sum_{i=1}^n (RSS_{m_{AP_i}} - RSS_{j_{AP_i}})^2} \quad (1)$$

1.2 Trägheitsnavigation mit INS-Algorithmus

Bei einem Trägheitsnavigationssystem oder engl. Inertial Navigation System (INS) werden mit den Beschleunigungssensoren Bewegungen erkannt und damit der zurückgelegte Weg abgeleitet sowie mit einem Kompass bzw. Magnetfeldsensor Fortbewegungsrichtungen bestimmt, um so auf die Änderung der Position zu schließen. Der in der Studie entwickelte und getestete INS-Algorithmus bedient sich eines einfachen Algorithmus zur Schritterkennung, der mit den Daten des Orientierungssensors von Android kombiniert wird. Der Orientierungssensor liefert Werte für die Fortbewegungsrichtung (Heading), Neigung (Pitch) und Verrollung (Roll), die von Android aus den Daten des Magnetfeldsensors und des Accelerometers hergeleitet werden.

Um einen Schritt zu detektieren, wurden die Daten des Beschleunigungssensors in der Z-Achse analysiert. Bei den Tests wurden die Smartphones wie in Abbildung 1 gezeigt gehalten. In Abbildung 2 ist ein Datensatz des Beschleunigungssensors für einen Testlauf beispielhaft dargestellt. An den mit roten Kreisen markierten Beschleunigungswerten befinden sich lokale Maxima, die jeweils einen Schritt darstellen. Diese wurden durch differenzieren der numerischen Werte ermittelt. Die Maxima werden als Schritt erkannt, wenn ein definierter Grenzwert von $11,54 \text{ m/s}^2$ überschritten wurde und zwischen diesen Werten eine definierte Zeitspanne von $0,4 \text{ s}$ liegt. Mit dem implementierten Schritterkennungsalgorithmus konnten Schritte auf eine Strecke von ca. 60 m auf $0,5$ Schritte genau erkannt werden.

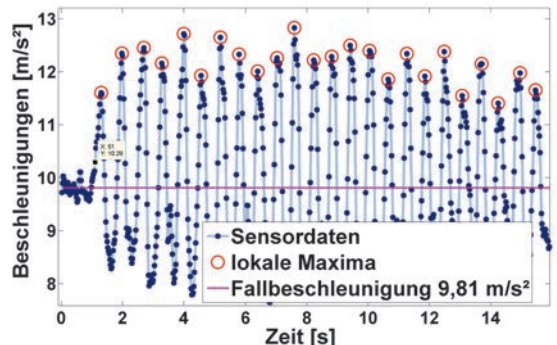


Abb. 2: Beispiel für Datensatz des Beschleunigungssensors in Z-Achse

Im INS-Algorithmus wird dann jedem erkannten Schritt der Richtungs-Wert des Orientierungssensors zugeordnet. Eine Änderung der Position erfolgt daher über einen detektierten Schritt und dessen Fortbewegungsrichtung. Die Größe der Änderung ist von einer zuvor kalibrierten Schrittweite abhängig.

Die Trägheitsnavigation weist jedoch zwei systembedingte Probleme auf. Erstens müssen eine Ausgangsposition bzw. ein Startpunkt und eine Startrichtung bekannt sein und zweitens kann die Änderung der Position nur mit einer bestimmten Genauigkeit berechnet werden. Das bedeutet, dass bei jeder Berechnung einer neuen Position ein Fehler gemacht wird, der zwar in erster Linie von der Genauigkeit der Sensoren bzw. des Algorithmus abhängig ist, aber der sich auch kumulativ auswirkt und daher mit der Zeit zu großen Positionsabweichungen führen kann. Ziel der Untersuchungen in dieser Studie ist es daher, diese Auswirkungen zu reduzieren bzw. zu kompensieren.

2. Zu Grunde liegende Daten und Algorithmen

2.1 Datenerfassungs- und Analysesystem (DAAS)

Um die gesteckten Ziele zu erreichen, benötigt man entsprechend gut eingestellte und getestete Algorithmen für das WLAN-Fingerprinting und die Trägheitsnavigation. Für die Datenerfassung wurde eine App für Androidbasis programmiert, die WLAN-Scans und INS-Sensordaten für eine bestimmte Position bzw. während eines Testlaufes aufzeichnet. Außerdem wurde ein umfangreiches MatLab-Framework entwickelt, welches zum Import und zur Auswertung der Messdaten dient. Mit diesem Framework und den gemessenen Daten als Basis können verschiedene Ansätze analysiert und Simulationen durchgeführt werden. Die Beta Version dieses Datenerfassungs- und Analysesystems (DAAS) steht online zur Verfügung¹⁾.

2.2 Testgebiete, Datensätze und Bewertung

Für die Tests wurden zwei Testgebiete bestimmt. Das Indoor-Testgebiet befindet sich vor und im neuen Elektrotechnischen Institutsgebäude der TU-Wien (EI-Testgebiet). Das Outdoor-Testgebiet wurde um den Häuserblock eines Studentenwohnheims in Meidling definiert (als Heim-Testgebiet bezeichnet). Als Testgeräte wurden ein Samsung Galaxy S2 und ein HTC EVO 3D verwendet.

1) https://drive.google.com/folderview?id=0B2sMc_nmy1A1aTFIZXZYb2ITTUk&usp=sharing

Zur Entwicklung des Fingerprinting-Algorithmus wurden an 72 Positionen Daten erfasst. Dabei wurden in der Trainingsphase 6.646 WLAN-Scans durchgeführt und gespeichert. Diese Daten wurden zur Erstellung der FPDB benutzt. Weitere 1.364 erfasste WLAN-Scans dienten zur Auswertung in der Positionierungsphase. Für die Analyse wurden insgesamt die Daten von 22 Testläufen aufgezeichnet. Als Bewertungskriterium für das Ergebnis des WLAN Fingerprintings mit den verschiedenen Ansätzen bzw. Einstellungen wurde das Verhältnis zwischen den korrekt zugewiesenen Positionen und deren Gesamtanzahl, die Erkennungsrate (ER), herangezogen.

2.3 Ergebnisse WLAN-Fingerprinting

Mit dem entwickelten Algorithmus konnten bei den ersten Untersuchungen im EI-Testgebiet alle untersuchten Positionen mit einer ER von vorerst nur 73,8% richtig zugewiesen werden. Für die FPDB wurden dabei die erfassten RSS-Werte einer Position mit dem arithmetischen Mittel berechnet. An jeder Position wurden die Scans dabei in vier um 90 Grad versetzten Orientierungen des Smartphones gemessen. Damit kann der Einfluss des menschlichen Körpers bei der Messung berücksichtigt werden. Diese Richtungen sind in Abbildung 3 und 4 für die jeweiligen Testgebiete mit blauen Pfeilen definiert. Wenn für einen AP kein RSS-Wert in einem vorgenommenen WLAN-Scan gemessen werden konnte, wurde ein minimaler Wert von -101 dBm für die Signalstärke angesetzt. Diese Vorgehensweise führt dazu, dass bei mehreren nicht messbaren RSS-Werten zu einem AP die Werte in der FPDB gegen diesen minimalen Wert konvergieren, was dann beim Fingerprinting einer Gewichtung gleichkommt.

Im Gebäude der Universität bieten die APs parallel mehrere WLAN-Netze (Multi-SSID) an (wie z.B. TU net, eduroam, usw.). Bei einem WLAN-Scan erhält man dann mehrere RSS-Werte für einen physikalisch einmal vorhandenen AP. Somit würden die RSS-Werte dieser Multi-SSID APs mehrfach gezählt werden, was zu einer höheren Gewichtung der Multi-SSID APs bei der Berechnung des euklidischen Abstandes führen würde. Durch das Erkennen dieser mehrfachen RSS-Werte und der Wahl eines RSS-Wertes konnten die ER im Mittel um 9,1 Prozentpunkte erhöht werden.

Mit einer zielgerichteten Gewichtung ausgewählter APs mit einem einfachen Optimierungsverfahren konnten um 4,1 Prozentpunkte höhere ER als vorhin erzielt werden. Die gesamte ER

konnte mit allen genannten Maßnahmen von 55,2 % auf 73,8 % erhöht werden.

Ein anderes Bild zeigte sich im Heim-Testgebiet. Hier konnten die WLAN-Scans der betrachteten Punkte bereits zu Beginn mit 97,4 % richtig zugeordnet werden. Bei diesem Outdoor-Testgebiet lagen die betrachteten Punkte, mit mindestens 10 m deutlich weiter voneinander entfernt als beim EI-Testgebiet, wo die Punkte oft nur 2 bis 3 m trennten. Außerdem bot keiner der APs Multi SSID an. Details zu diesen Ergebnissen sind in [4] zu finden.

2.4 Ergebnisse INS-Algorithmus

Der entwickelte INS-Algorithmus kombiniert den Schritterkennungsalgorithmus mit den Daten des Orientierungssensors. Zwei Beispiele für Testläufe, bei denen der INS-Algorithmus zur Ermittlung der zurückgelegten Strecke angewandt wurde, sind in den Abbildungen 3 und 4 dargestellt. Die ermittelten Schritte des INS-Algorithmus sind als violette Punkte und die Referenzschritte als grüne Punkte visualisiert. Diese wurden anhand der erkannten Schritte und der passierten Referenzpunkte bestimmt. Diese Referenzpunkte sind als neon-grüne Sterne dargestellt. Das Passieren dieser Hilfspunkte wurde mit der entwickelten App per manueller Eingabe durch den Nutzer festgehalten. Zum Vergleich sind in Abbildung 3 die erfassten GPS-Koordinaten des Testlaufs zu den Schrittzeitpunkten in blau eingetragen. Bei diesem Testlauf wichen die mit dem INS-Algorithmus ermittelten Positionen um 4,9 m und die mit GPS ermittelten um 15,5 m von den Referenzpunkten ab.

Abbildung 4 zeigt die Ergebnisse eines Testlaufes im Institutsgebäude. Wie auch in Abbildung 3 sind hier die errechneten Koordinaten des INS-Algorithmus den Referenzschritten gegenübergestellt. Bei diesem Indoor-Test waren keine GPS Signale verfügbar. Die Position des Endpunkts (siehe Abbildung 4) konnte bei diesem Testlauf auf 3,7 m genau bestimmt werden. Im Mittel konnten die Positionen auf 5,4 m genau mit dem INS-Algorithmus berechnet werden. Zum Vergleich, bei GPS lag die Abweichung bei 16,7 m vor dem Betreten des Gebäudes, wenn eine Position zur Verfügung stand, was bei den Indoor-Testläufen nicht der Fall war.

3. Das Konzept der intelligenten Check-Points (iCPs)

Führt man sich vor Augen, wie man in einen bestimmten Raum eines Gebäudes gelangt, so müs-

sen gewisse Wegpunkte passiert werden, um dort hinzugelangen. Zuerst wird ein Gebäudeeingang gewählt. In vielen Fällen wird man dort eine Aula oder Ähnliches betreten. Um in einen anderen Stock zu gelangen, muss entweder der Lift oder die Treppe benutzt werden. Bevor man den Raum betreten kann, muss erst ein Gang gewählt und abgegangen werden. Diese Wegpunkte stehen in Abhängigkeit zueinander und müssen nach einer logischen Abfolge abgeschrieben werden, um von A nach B zu gelangen. Türen, Stiegen und Gänge und dergleichen können auch als Knotenpunkte oder Vektoren eines Graphen betrachtet werden.



Abb. 3: Trägheitsnavigation mit INS-Algorithmus im Outdoor-Testgebiet



Abb. 4: Trägheitsnavigation mit INS-Algorithmus im Indoor-Testgebiet

Je weiter man in das Gebäude vordringt, eine umso tiefere Ebene des Graphen erreicht man. Aus diesen Ebenen lassen sich auch Kategorien ableiten, die bereits eine einfache logische Abfolge beschreiben können. Bei der von uns gewählten Logik befinden sich Gebäudeeingänge auf Ebene 0. Die Zugänge zu Treppenhaus, Lift und Aula auf Ebene 1. Die Türen zu den Gängen bzw. den Liften in den weiteren Stockwerken auf weiteren Ebenen. In einer Subebene befinden sich alle Türen zu den Räumen eines Stockwerks. Dies ist eine einfache mögliche Darstellungsform, die die Zusammenhänge von Gängen und Räumen in einem Gebäude beschreibt. Um nun von A nach B gelangen zu können, müssen diese Ebenen nacheinander passiert werden.

Bei der Wahl dieser sog. Check-Points sind die Strukturen des Gebäudes nutzbar, wie z. B. Wände, Ecken oder Türen, die auch die Signalstärken zu den vorhandenen APs beeinflussen können. Es müssen immer wieder unterschiedliche Engpässe passiert werden, um voranzukommen. Außerdem können diese Wegpunkte oft nur in einer bestimmten Richtung passiert werden. Auch diese Information kann bei der Navigation genutzt werden.

Beim Fingerprinting werden die Signalstärken der APs an bekannten Punkten in der Trainingsphase gemessen, wobei die Messpunkte in der Regel in einem gleichmäßigen Raster angelegt werden. Diese Vorgehensweise ist sehr arbeitsintensiv, da für eine entsprechende Positionierungsgenauigkeit die Rasterpunkte sehr dicht gewählt werden müssen. Der neue Ansatz verfolgt das Ziel, die notwendige Anzahl von Messpunkten entscheidend zu reduzieren. Es werden nun die Messpunkte auf intelligente Weise ausgewählt, damit diese als gut unterscheidbare Knotenpunkte verwendet werden können. Diese intelligenten Check-Points (kurz iCPs) werden dann für die Navigation durch das Gebäude genutzt.

4. Dynamische Erkennung des Passierens der iCPs

Das Konzept der iCPs sieht vor, zu erkennen, wann diese erreicht bzw. ob diese passiert wurden und ob eine berechnete Position korrigiert werden muss. Der Zeitpunkt des Passierens wird mithilfe von WLAN-Fingerprinting erkannt. Die Grundidee liegt darin, die infrage kommenden in der Nähe befindlichen iCPs mithilfe der Abfolgelogik zu ermitteln. Anhand der gespeicherten RSS-Vektoren der iCPs und den zuletzt gemessenen WLAN-Scans, kann der euklidische Abstand (Abstandswert) der

RSS-Werte zu unterschiedlichen Zeitpunkten berechnet werden. Der Ansatz geht davon aus, dass zum Zeitpunkt des Passierens des beobachteten iCPs der errechnete euklidische Abstand minimal ist.

Wird also ein iCP_a , dessen ermittelte RSS-Werte durch RSS_{iCP_a,AP_k} dargestellt sind, über einen bestimmten Zeitraum beobachtet, kann für jeden Zeitpunkt t , an dem RSS-Werte durch einen WLAN-Scan erfasst wurden, ein Abstandswert d_t ermittelt werden. Anhand von Gleichung 1 erhält man die Abstandswerte d_t für den Zeitraum $[t_1, \dots, t_T]$ wie in Gleichung 2 erläutert.

$$\begin{bmatrix} d_{t_1} \\ \vdots \\ d_{t_T} \end{bmatrix} = \begin{bmatrix} \sqrt{\sum_{k=1}^n (RSS_{m_{t_1,AP_k}} - RSS_{iCP_a,AP_k})^2} \\ \vdots \\ \sqrt{\sum_{k=1}^n (RSS_{m_{t_T,AP_k}} - RSS_{iCP_a,AP_k})^2} \end{bmatrix} \quad (2)$$

So ergeben sich über den beobachteten Zeitraum, entsprechend der durchgeführten WLAN-Scans, T Abstandswerte zu dem iCP_a . Abbildung 5 zeigt einen Ausschnitt des Testlaufs für iCP_3 im Heim-Testgebiet. Der iCP_3 ist in Abbildung 3 abgebildet. Die berechneten Abstandswerte d_t sind in der oberen grünen Kurve dargestellt. Die untere blaue Kurve stellt die Abstandswerte d_t geglättet dar. Diese gleitenden Mittelwerte wurden für unsere Tests mithilfe der MatLab Funktion `smooth()` berechnet. Der Zeitpunkt, an dem der iCP_a passiert wurde, ist mit einer roten vertikalen Linie markiert. Der anhand des Minimalwertes berechnete Zeitpunkt ist grün dargestellt. Durch das Glätten der Abstandswerte

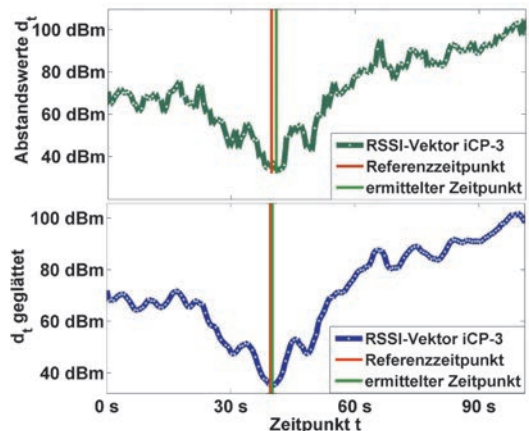


Abb. 5: Euklidische Abstände der erfassten RSS-Werte zu iCP_3 (Outdoor-Testgebiet) bei einem Testlauf

konnten die Zeitpunkte des Passierens tendenziell genauer erkannt werden. In Abbildung 5 ist zu sehen, dass der berechnete Zeitpunkt und der Referenzzeitpunkt durch das Glätten besser übereinstimmen. Da Positionsveränderungen anhand von Schritten ermittelt wurden, wurde die Abweichung beim Erkennen des Passierens der iCPs ebenfalls in Schritten berechnet und verglichen. Für eine Echtzeit-App müssten die erfassten Daten gepuffert werden, um nach dem Passieren eines iCPs die Position nachkorrigieren zu können.

Bei der Testreihe zur Untersuchung der dynamischen Erkennung der iCPs konnten die iCPs im Heim-Testgebiet im Mittel auf 1,0 Schritte (mit dem HTC Smartphone mit Glättung) genau erkannt werden. Indoor konnten die iCPs im EL-Testgebiet durchschnittlich auf 3,0 Schritte (HTC mit Glättung) genau erkannt werden. Gut ließen sich jene iCPs erkennen, wo sich, durch die vorhandenen baulichen Strukturen, die RSS-Werte beim Passieren deutlich veränderten. So ließen sich z. B. jene iCPs bei den Eingängen bzw. im Windfang des Eingangs am besten erkennen. Mit dem Samsung Galaxy S2 Smartphone war das Erkennen der iCPs deutlich ungenauer. Mit diesem Gerät konnten die iCPs im Mittel nur auf 7,2 Schritte erkannt werden. Der Grund dafür liegt darin, dass ein Scan mit dem Galaxy S2 im Schnitt 5-mal (ca. 3,5 s) länger dauert als mit dem HTC (ca. 0,7 s).

5. Korrektur der Position durch iCPs

Mit dem von uns vorgeschlagenen Fingerprinting Ansatz zur Erkennung des Passierens von iCPs war es uns möglich, diese im Schnitt auf 3,0 Schritte genau zu erkennen. Bei der verwendeten Schrittweite von 0,7 m entspricht das einer Genauigkeit von 2,1 m. Die erkannten iCPs können somit genutzt werden, um die mit dem INS-Algorithmus berechneten Positionen zu korrigieren. Beim ersten Test im Outdoor-Bereich werden dazu wenige einfache Regeln angewendet. Zunächst wird jedem iCP ein Einflussbereich zugewiesen, wobei diese Bereiche, je nach iCP, mit einem Radius von 4 bis 5,5 m um den iCP gebildet werden. Fällt eine mit dem INS-Algorithmus errechnete Position in den Einflussbereich, wird eine bestimmte Anzahl der erfassten WLAN-Scans vor Erreichen und Verlassen des Einflussbereichs mit dem RSS-Vektor des iCP verglichen und die Abstandswerte d_t werden berechnet. Für den minimalen Abstand wird somit ein Zeitpunkt ermittelt an dem der iCP vermutlich passiert wurde. Der Zeitpunkt liegt



Abb. 6: Beispiel für die Korrektur der Positionen durch Erkennen der iCPs

zwar einige Schritte zurück, kann aber zum Korrigieren der aktuell berechneten Position verwendet werden. Dann wird die Position zum ermittelten Zeitpunkt auf die gespeicherte Position der iCPs gesetzt. Die mit dem INS-Algorithmus berechneten Positionsveränderungen werden dann von dieser Position fortgeführt. Abbildung 6 zeigt, wie die errechnete Position des INS-Algorithmus auf diese Weise korrigiert werden, wobei die korrigierten Positionen mit roten Linien eingetragen sind.

Da die iCPs auf Positionen definiert sind, die mit hoher Wahrscheinlichkeit von Fußgängern benutzt werden, gelingt es bereits mit diesen einfachen Regeln das Abdriften des INS-Algorithmus zu verhindern. Eine weitere Regel könnte sein, einen Grenzwert für den berechneten minimalen Abstandswert der RSS-Werte einzuführen.

5.1 Erkennung des ersten iCPs

Für die Navigation mit dem INS-Algorithmus muss die Startposition bekannt sein. Die Grundidee des von uns benutzten Ansatzes zur Berechnung der Startposition liegt darin, das erstmalige Passieren eines iCP zu erfassen und aus den aufgezeichneten Sensordaten auf die Startposition zurückzuschließen. Da die Anzahl der iCPs sehr hoch sein kann, werden nur die iCPs, die sich in der Nähe der vermuteten bzw. zuletzt bekannten Position befinden, betrachtet.

Um eine Position abschätzen zu können, wurden dafür auch die GPS-Koordinaten zum Startzeitpunkt des Testlaufes verwendet. Dann werden die k geografisch nächsten iCPs herangezogen und deren RSS-Werte mit den erfassten verglichen. Gewählt wird jener iCP, bei dem als erstes ein Minimum zu den erfassten RSS-Werten er-

mittelt wurde. Aus dem erfassten Zeitpunkt und der bekannten Position des iCP kann mithilfe des INS-Algorithmus dann die Startposition berechnet werden.

Abbildung 7 zeigt, wie nach dem Passieren des ersten iCPs vom GPS-Startpunkt (blauer Stern) ein neuer Startpunkt (Stern in Magenta) ermittelt wurde. Der ermittelte Startpunkt weicht bei diesem Test um 3,0 m vom Referenzstartpunkt (türkiser Stern) ab, während der GPS-Startpunkt um 14,2 m abweicht.

In Abbildung 8 ist ein Testlauf dargestellt, bei dem das Elektrotechnische Institutsgebäude betreten wird. Dabei wird der iCP1 als gewählter Eingang erkannt und so eine Startposition ermittelt. So ist es möglich, die Startposition auf 2,0 m genau zu bestimmen. Mit GPS war das nur auf 13,4 m möglich.

Für das EI-Testgebiet wurden die iCPs in verschiedene Ebenen eingeteilt (vgl. Abschnitt 3), wobei jene iCPs, die sich an den Eingängen des Gebäudes befinden, der gleichen Ebene zugewiesen wurden. Der Algorithmus betrachtet dann nur die k nächsten auf der Eingangsebene befindlichen iCPs, um zu erkennen, wo und wann das Gebäude betreten wird. Über alle Testläufe betrachtet konnte die Startposition auf 3,6 m genau erkannt werden. Mit GPS hingegen wich die Startposition im Mittel um 20,0 m ab.

5.2 iCP-INS-Algorithmus

Der iCP-INS-Algorithmus kann also auch die logische Abfolge der iCPs berücksichtigen. Die iCPs können noch intelligenter genutzt werden, indem beim Passieren bestimmter iCPs die Schrittweite korrigiert oder die Fortbewegungsrichtung kalibriert wird. Die Schrittweite wird z. B. beim Treppensteigen auf das entsprechende Schrittmaß gesetzt. Das Passieren von Türen und Gängen, deren vordefinierte Richtung bekannt ist, wird genutzt, um einen Kalibrierungswert für die Fortbewegungsrichtung zu bestimmen. Außerdem kann die Korrektur der Position auf die vorangegangenen Schrittpositionen verteilt werden, um ein kontinuierliches Abbild der Strecke zu erhalten.

5.2.1 Korrektur des Fortbewegungsrichtung

Die vom Orientierungssensor von Android bereitgestellte Fortbewegungsrichtung ist abhängig von den Daten des Magnetfeldsensors. Gerade in Gebäuden können starke elektromagnetische Felder den Sensor beeinflussen. In Abbildung 10 ist ein Testlauf entlang des Ganges im dritten



- * Referenzpunkte
- * ermittelter Startpunkt
- GPS-Aufzeichnung
- iCPs
- korrektur Startpos.
- Referenzschritte
- * Referenz-Startpunkt
- iCP-INS-Algorithmus
- * GPS-Startpunkt

Abb. 7: Beispiel zur Rückrechnung der Startposition im Outdoor-Testgebiet



- iCPs
- * Referenzpunkte
- Referenzschritte
- * ermittelter Startpunkt
- * Referenz-Startpunkt
- iCP-INS-Algorithmus
- * GPS-Startpunkt

Abb. 8: Beispiel zur Rückrechnung der Startposition im Indoor-Testgebiet

Stock des EI-Testgebiets dargestellt. Betrachtet man die Ergebnisse des INS-Algorithmus, ist zu sehen, dass die vom Sensor erhaltene Fortbewegungsrichtung bereits zu Beginn entlang weniger Schritte stark vom wahren Wert abweicht (siehe rechts oben). Genau an dieser Stelle befindet sich einer der großen Kabelschächte des Gebäudes. In Abbildung 9 ist zu sehen, dass die Fortbewegungsrichtung nach dem Betreten des

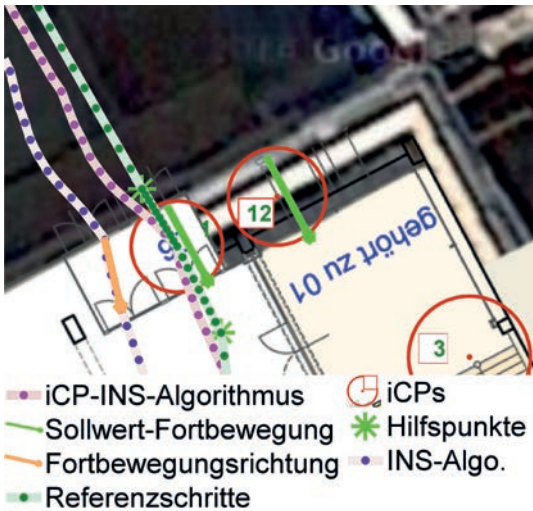


Abb. 9: Beispiel für Korrektur der Fortbewegungsrichtung beim Betreten des Gebäudes bzw. Passieren eines iCPs

Gebäudes, eine höhere Abweichung hat als zuvor im Außenbereich.

Der Ansatz geht weiter davon aus, dass aufgrund von baulichen Gegebenheiten manche iCPs nur in einer bestimmten Richtung passiert werden können und diese Information genutzt werden kann, um einen Korrekturwert zu berechnen. Für die entsprechenden iCPs wurden diese Sollwerte für die Fortbewegungsrichtung definiert,

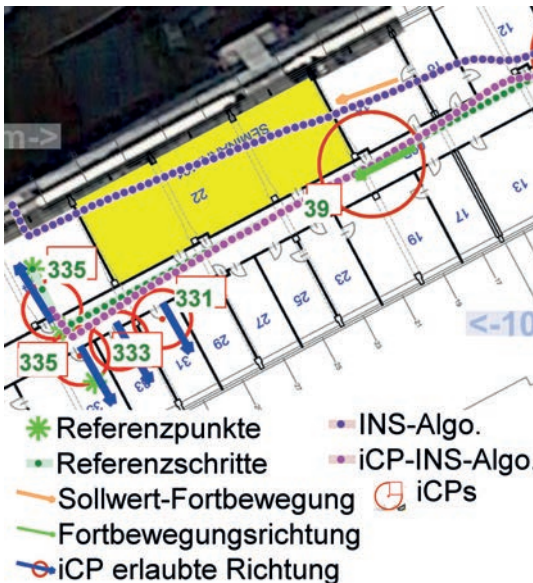


Abb. 10: Beispiel für die korrigierte Fortbewegungsrichtung bei iCPs am Gang und erlaubte Richtungsklassen

sowie eine Anzahl von Schritten, für die diese Richtungen vor und nach den iCPs gelten muss. Aufgrund des erfassten mittleren Werts für die Richtung und des Sollwertes wurde ein Korrekturwert ermittelt. Dieser Ansatz wurde bei den iCPs an den Eingängen und auf den Gängen verwendet. Der Korrekturwert wird so lange beibehalten, bis ein neuer Korrekturwert bei einem anderen iCP errechnet wird. In Abbildung 9 wird beim Eingang mit dem iCP1 und in Abbildung 10 bei dem iCP39 ein Korrekturwert ermittelt. Die ermittelten Trajektorien mit dem iCP-INS-Algorithmus sind in Magenta, die ohne Korrektur violett eingezeichnet. Die ermittelte mittlere Fortbewegungsrichtung ist mit orangen Pfeilen visualisiert.

5.2.2 Anpassung der Schrittweite

Bei der Schritterkennung wurde vorerst mit konstanten Schrittweiten für einen Nutzer gearbeitet, die am Beginn durch eine Kalibrierung definiert wurde. Bei Treppen ist jedoch die Weite der Schritte durch den Abstand der Treppen definiert. Daher passt der iCP-INS-Algorithmus diese Schrittweite dort an.

Vergleicht man in Abbildung 11 den INS-Algorithmus mit dem iCP-INS-Algorithmus, so ist zu sehen, dass die Schrittweite auf 0,27 m für 16 Schritte angepasst wurde, nachdem der iCP3 passiert wurde. Dies entspricht dem Treppenabstand und der Anzahl an Stufen bis zum iCP9.

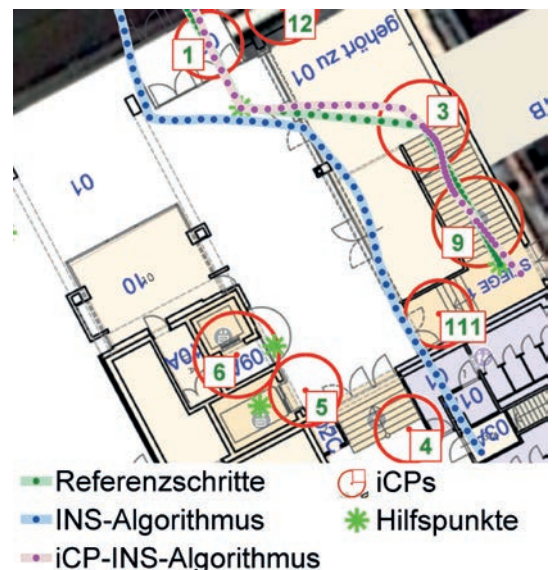


Abb. 11: Beispiel Korrektur der Schrittweiten nach erkanntem iCP vor einer Treppe

Wird dieser passiert, wird ebenfalls die Schrittweite für 5 weitere Stufen angepasst. Vergleicht man die violette Strecke, die mit dem INS-Algorithmus ermittelt wurde, so ist zu sehen, dass die Schrittweite im Bereich der Stiege zu weit wäre. Bei den Tests lag die optimale Schrittweite mit der gleichen Testperson Outdoor bei 0,67 m und Indoor bei 0,65 m. Die Differenz erklärt sich wahrscheinlich dadurch, dass im Außenbereich automatisch etwas größere Schritte gemacht werden, als in einem Gebäude. Daher ist die Anpassung der Schrittweite, sobald man ein Gebäude betritt, ebenfalls sinnvoll.

5.2.3 Erlaubte Richtungen

Die Tatsache, dass bestimmte iCPs nur entlang einer bestimmten Richtung passiert werden können, wurde noch in einer anderen Weise genutzt. So können z. B. die Türen zu den Räumen des Gebäudes nur entsprechend deren vorgegebener Richtung passiert werden.

Für die Vermessung der iCPs werden vier Hauptrichtungen des Smartphones (vgl. Abbildung 4) definiert, in denen jeweils RSS-Werte ermittelt werden. Diese Orientierungen beziehen sich auf die wesentlichen möglichen Richtungen im Gebäude vorgegeben durch Gänge und Türen bzw. im Outdoor-Bereich entlang von Straßen. Beim Erfassen der RSS-Werte wurde somit die Fortbewegungsrichtung erfasst und manuell einer der vier Richtungsklassen zugewiesen. Entsprechend dieser Daten wurden Bereiche definiert, um alle anderen Bewegungsrichtungen einer der vier Klassen zuweisen zu können. Somit beschreibt eine Richtungsklasse einen Wertebereich von Fortbewegungsrichtungs-Werten, die für eine der definierten Richtungen erfasst wurden. Bei einem idealisierten Modell erhält man so eine Normalverteilung deren Erwartungswert der definierten Richtung entspricht.

Für die Testläufe wurde jedem Schritt eine Richtungsklasse zugewiesen. Dies wird von dem iCP-INS-Algorithmus verwendet, um weitere Einschränkungen für das Erkennen eines iCP zu machen. Wie in Abbildung 10 zu sehen ist, müssen die Türen zu den Büros (Raum 31, 33 und 35) und zum Zeichensaal 1 (Raum 34) nach bestimmten Richtungskategorien durchschritten werden. Diese sind durch die blauen Pfeile symbolisiert. Das hat den Sinn, dass iCPs nicht erkannt werden, wenn man nur den Gang entlanggeht. Daher wird die Einschränkung gemacht, dass die iCPs 335,

333, 331 und 334 nur in den Richtungen entsprechend der Türen passiert werden können.

6. Ergebnisse, Konklusion und Ausblick

Betrachtet man alle Testläufe, weicht der iCP-INS-Algorithmus um 1,9 m und der INS-Algorithmus um 4,5 m von den Positionen der Referenzschritte ab. Detaillierte Ergebnisse sind in der Tabelle 1 festgehalten. Die Endposition konnte mit dem iCP-INS-Algorithmus durchschnittlich auf 1,5 m und mit dem INS-Algorithmus auf 5,7 m genau erkannt werden. Detaillierte Angaben zu den Testläufen und beschriebenen Ergebnissen sind in [5] zu finden.

Wie an den Resultaten der Testläufe zu sehen ist, konnten mit dem vorgestellten Ansatz gute Ergebnisse bei der Indoor-Navigation erzielt werden. Bei den Testläufen wurden alle iCPs korrekt erkannt und so war es möglich, für alle Testläufe die korrekte Endposition bzw. den richtigen Raum zu erkennen. Mit dem vorgestellten iCP-INS-Algorithmus konnte durch das Korrigieren der Positionen das Abdriften des INS-Algorithmus verhindert werden.

Der iCP-INS-Algorithmus stellt einen möglichen Ansatz dar, wie die Messdaten von GPS, Bewegungssensoren und vom WLAN-Modul kombiniert werden können, um diese zusammen zur Navigation zu nutzen. Gezeigt wurde auch, wie ortsgebundene Informationen genutzt werden können,

Testlauf	Mittlere Abweichung		
	GPS	INS	iCP-INS
TL-H7	10,1 m	6,4 m	3,0 m
TL-H9	11,6 m	3,8 m	0,7 m
TL-H10	12,9 m	4,4 m	1,8 m
TL-EI1	34,5 m	2,9 m	2,3 m
TL-EI2	20,7 m	6,0 m	4,0 m
TL-EI3	16,4 m	3,2 m	1,9 m
TL-EI4	12,8 m	4,3 m	1,5 m
TL-EI5	14,7 m	4,2 m	1,9 m
TL-EI6	n. a.*	4,8 m	1,5 m
TL-EI7	n. a.*	3,8 m	1,1 m
TL-EI8	n. a.*	5,5 m	1,3 m
	16,7 m	4,5 m	1,9 m

* not available

Tab. 1: Übersicht der mit ICP-INS-ALGORITHMUS untersuchten Testläufe

um die Navigation weiter zu verbessern. So wurden die Schrittweiten bei den Treppen angepasst und die vordefinierte Richtung der Eingänge und des Ganges für Richtungskorrekturen berücksichtigt. Das machte es möglich, Abweichungen in Richtung und Schrittweite auszugleichen. Im Vergleich zum Map Matching der Trajektorie auf eine digitale Karte der Umgebung ist keine genaue Karteninformation notwendig. Die Karte dient rein zur Visualisierung der Trajektorie und der Referenzpunkte. Es genügt wenige Regeln und den Maßstab zu definieren, um die Schritte eines Nutzers sehr genau darstellen zu können.

Mit dem iCP-Ansatz konnte in allen Fällen der richtige Raum erkannt werden. In früheren Arbeiten im EI-Testgebiet lag die Erkennungsrate bei nur rund 23 % (siehe [6]). Ein weiterer Vorteil im Vergleich zum herkömmlichen WLAN Fingerprinting ist die erhebliche Reduzierung des Arbeits- und Zeitaufwandes in der Trainingsphase. Normalerweise werden beim Fingerprinting Referenzpunkte in einem regelmäßigen Raster eingemessen. Abhängig von der Rasterweite bedeutet dies einen hohen Arbeitsaufwand für die Kalibrierung. Bei unserem Ansatz gibt es nur ausgewählte Referenzpunkte, die iCPs, an denen Trainingsmessungen ausgeführt werden müssen. Auch in der Positionierungsphase wird die notwendige Rechenzeit zur Ermittlung der Positionen deutlich reduziert, da nur die iCPs in unmittelbarer Nähe zur aktuellen Position des Nutzers getestet werden müssen. Damit kann der notwendige Arbeitsaufwand um einen Faktor 4 bis 5 reduziert werden. Im 3. Stock des EI-Testgebiets wären anstelle von 75 Referenzpunkten, die in vorgehenden Untersuchungen [6] eingemessen wurden, nur mehr 17 iCPs notwendig, wenn diese an Zugängen zum Korridor und nach jeder Tür innerhalb eines Büros angeordnet werden. Wenn der INS-Algorithmus die kontinuierlichen Positionen des Nutzers genau genug liefert, kann eine weitere Reduktion der iCP vorgenommen werden, z. B. mehrere iCPs entlang des Gangs und nicht in jedem Raum bei der Tür.

Der Ansatz mit den iCPs wurde vorerst mit deterministischen Algorithmen realisiert. Die Entscheidung, ob ein iCP als passiert erkannt wird, ist abhängig von bestimmten Bedingungen. Wird ein iCP falsch oder nicht erkannt, kann das zum Versagen des Ansatzes führen. Daher wäre es vermutlich sinnvoll, einen stochastischen Ansatz mit den hier vorgestellten iCPs zu realisieren. So könnten man die verwendeten Informationen, wie euklidischer Abstand zu RSS-Werten des iCPs,

Richtung des Passierens, die aktuell errechnete Position und die logischen Zusammenhänge der iCPs nutzen, um einen Wahrscheinlichkeitswert für das Passieren eines iCPs zu ermitteln.

Es gibt noch weitere Sensordaten, die genutzt werden könnten. Z. B. könnte versucht werden, mithilfe der Daten des Beschleunigungssensors und des Barometers zu erkennen, ob und zu welchem Zeitpunkt die Stiege oder der Lift benutzt werden. Während der Trainingsphase beim Vermessen der iCPs können die Messdaten des Magnetfeldsensors zusätzlich erfasst werden und zur Bestimmung von Korrekturwerten für die Fortbewegungsrichtung dienen, denn durch die vier definierten Orientierungen bei der Kalibrierung steht einem errechneten Orientierungswert ein Referenzwert bei der Vermessung gegenüber. Zusätzlich muss man auch bei nur zwei möglichen Fortbewegungsrichtungen (z. B. entlang eines Korridors) nicht in allen vier Orientierungen die Signalstärken messen. Dadurch verringert sich der Arbeitsaufwand in der Trainingsphase nochmals.

Der neue iCP-Ansatz konnte zwei wesentliche Nachteile des WLAN Fingerprintings beseitigen. Einerseits wird der hohe Arbeitsaufwand bei der Kalibrierung entscheidend reduziert und andererseits die geringe Positionierungsgenauigkeit signifikant verbessert. Weiteres Verbesserungspotenzial besteht höchstwahrscheinlich darin, wenn die iCPs anhand eines stochastischen Modells erkannt werden, bei dem neben dem WLAN Signalstärkemessungen weitere Smartphonesensoren eingebunden werden. Vorstellbar sind auch selbstlernende iCPs, die z. B. anhand eines neuronalen Netzwerkes trainiert werden.

Referenzen

- [1] *Retscher, G.; Hecht, T. (2012): Investigation of Location Capabilities of Four Different Smartphones for LBS Navigation Applications.* In: Rizos, C.; Dempster, G. A.; Li B.; Gallagher T. (Hrsg.): IEEE Xplore, 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN'12), Sidney, Australien, 13.-15. November, 6 S.
- [2] *Mok, E.; Retscher, G.; Wen, C. (2012): Initial Test on the Use of GPS and Sensor Data of Modern Smartphones for Vehicle Tracking in Dense High Rise Environments.* In: Kuusniemi H. (Hrsg.): IEEE Xplore, 2012 Ubiquitous Positioning, Indoor Navigation, and Location Based Service (UPINLBS'12). Helsinki, Finnland, 3.-4. Oktober, 7 S.
- [3] *Bahl, P.; Padmanabhan, V. N. (2005): RADAR: An In-Building RF-Based User Location and Tracking System.* In: Omar A. M. Aly, A. S. Omar (Hrsg.): 2nd Workshop on Positioning, Navigation and Communication ULTRA-WIDEBAND EXPERT TALK (UET'05), Magdeburg, Deutschland, S. 775-784.

- [4] Retscher, G.; Hofer, H. (2015): A Novel Approach for Wi-Fi Fingerprinting Using Logical Sequences of Intelligent Checkpoints. IGNSS 2015 Conference, Surfers Paradise, Gold Coast, Queensland, Australien, 14.-16. Juli, 16 S.
- [5] Hofer, H. (2015): Kombinierte Indoor/Outdoor Positionierung mit Smartphones. Diplomarbeit, Department für Geodäsie und Geoinformation und Fakultät für Informatik, Technische Universität Wien.
- [6] Retscher, G. (2012): Wi-Fi Positioning with Smartphones. 9th International Symposium on Location-Based Services LBS 2012, Oktober 16-18, München, Deutschland, 9 S.

Anschrift der Autoren

Dipl.-Ing. Hannes Hofer, Department für Geodäsie und Geoinformation, Forschungsgruppe Ingenieurgeodäsie, Technische Universität Wien, Gußhausstraße 27-29 E120/5, 1040 Wien.

E-Mail: h.hannes@gmx.at

A. o. Univ.-Prof. Dr. Günther Retscher, Department für Geodäsie und Geoinformation, Forschungsgruppe Ingenieurgeodäsie, Technische Universität Wien, Gußhausstraße 27-29 E120/5, 1040 Wien.

E-Mail: guenther.retscher@tuwien.ac.at

vgi



International Map Year (IMY) is a worldwide celebration of maps and their unique role in our world. It's organized by the International Cartographic Association (ICA) and supported by the United Nations (UN).

mapyear.org

