

Sachbericht
Projekt Per Pedes Routing
Teilprojekt Darmstadt
Förderkennzeichen 19F2001B

Problemstellung und Bedeutung

Ganz allgemein wird unter Routing hier die Suche nach möglichst guten Reisewegen von einem Startpunkt zu einem Zielpunkt verstanden. Darunter fallen beispielsweise Navis im Auto, also Suche nach Fahrtrouten im Straßenverkehrsnetz, aber etwa auch Fahrplanauskunftssysteme, das heißt, Reisewege mit öffentlichen Verkehrsmitteln wie Bahn und Bus. Zu allen bislang relevanten Arten von Routing existiert umfangreiche Forschung.

Trotz seiner offenkundig hohen gesellschaftlichen Relevanz finden sich nur punktuelle Forschungsergebnisse zu Routing für Personen mit Einschränkungen. Hier setzen wir an mit dem Ziel, die Problematik möglichst allgemein zu erfassen, zu modellieren und in geeignete Routing-Algorithmen zu übersetzen. Das Projekt Per Pedes Routing konzentrierte sich auf reine Fußwege, war aber von Anfang an auf die Entwicklung von Algorithmen angelegt, die an erweiterte Szenarien wie etwa multimodale Reiseketten angepasst werden können, also Einbeziehung von öffentlichen Verkehrsangeboten (Bahn, Bus etc.) und Individualverkehr (eigenes KFZ, Taxi, Car-Sharing etc.). Eine solche Erweiterung geht aber über Per Pedes Routing hinaus und wäre Thema eines weiteren Projektes.

Das konkrete algorithmische Problem, das im Rahmen des Projektes Per Pedes Routing an der TU Darmstadt bearbeitet wurde, besteht also darin, möglichst gute Fußwege von einem Startpunkt (z.B. von zuhause) zu einem Zielpunkt (z.B. Bushaltestelle) zu finden. Das Kriterium für die Güte eines Fußwegs setzt sich zusammen aus (1) seiner Länge bzw. geschätzten Dauer und (2) seiner Beschwerlichkeit: Für Personen mit Einschränkungen ist beispielsweise eine Treppe ein Hindernis, das eher vermieden werden sollte, solange der dafür notwendige Umweg nicht zu lang wird, bei schwerer Einschränkung muss die Treppe in jedem Fall umgangen werden.

Die wesentlichen Kriterien für die Güte eines algorithmischen Ansatzes sind (1) Güte der berechneten Fußwege wie im letzten Absatz skizziert und (2) die Antwortzeiten für Suchanfragen.

Eine solche Fußwegsuche muss berücksichtigen, in welcher Art und wie schwer Nutzer individuell eingeschränkt sind. Im Beispiel vom letzten Absatz ist der Umweg, den Nutzer zur Vermeidung einer Treppe zu tolerieren bereit sind, abhängig davon, wie beschwerlich diese Treppe für den Nutzer individuell ist. Daher muss der Algorithmus auf eine möglichst große Bandbreite von individuellen Nutzerprofilen anwendbar sein.

Voraussetzungen

Die Arbeitsgruppe Algorithmik der TU Darmstadt, in der dieses Teilprojekt von Per Pedes Routing bearbeitet wurde, hat im Laufe der letzten ca. zehn Jahre ein multimodales Fahrplanauskunftssystem namens MOTIS entwickelt, in das die barrieresensible Fußwegsuche aus Per Pedes Routing integriert werden konnte und sollte und so in einem weiteren Schritt mit öffentlichen Verkehrsangeboten und privaten Verkehrsmitteln zu multimodalen Reiseketten kombiniert werden kann.

Überblick

Der nun folgende Überblick über den weiteren Sachbericht ist nach der Empfehlung für die Struktur von Sachberichten im zugehörigen Merkblatt strukturiert.

Ziele und Schwerpunkte

Die Schwerpunkte der Arbeiten, die an der Technischen Universität Darmstadt im Rahmen des Per Pedes Routing Projekts durchgeführt wurden, lagen primär in der Konzeption und Implementierung prototypischer Softwarekomponenten. Dies umfasst die Konzeption und Umsetzung aller Komponenten im Bereich der Datenhaltung, des algorithmischen Routing-Kerns und der Erstellung von geeigneten Benutzeroberflächen (sowohl als Webseite als auch als mobile Android Applikation).

Erstellung und insbesondere Evaluierung dieser Softwarekomponenten war das Ziel. Dieses Ziel wurde vollumfänglich erreicht.

Wie im Abschnitt Validierung im Hauptteil unten näher ausgeführt, berechnet unsere Software optimale Lösungen unter praxisnahen Antwortzeiten und hat damit ihre prinzipielle Praxistauglichkeit unter Beweis gestellt.

Die Hochschule Fresenius (Prof. Haas) war unser primärer Kooperationspartner im Rahmen des Projekts. Wesentliches Ziel der Kooperation war die gemeinsame Definition von möglichst generischen, vereinheitlichenden Modellen für eine möglichst große Vielfalt an (1) Arten von Barrieren mit ihren jeweiligen Eigenschaften, (2) Formen von Einschränkungen von Nutzern und (3) der Verknüpfung von (1) mit (2). Dafür mussten die Erkenntnisse aus dem Bereich Gesundheitsforschung (Hochschule Fresenius) in Einklang gebracht werden mit den Möglichkeiten und Grenzen der Algorithmik (TU Darmstadt). Die an der TU Darmstadt entwickelte Software basiert darauf.

Die Software wurde im Laufe des Projekts erfolgreich zu einem vollständigen System von der Datenerfassung, Datenaufbereitung und Datenhaltung über den Routing Algorithmus bis hin zu Bedienoberflächen (mobil und web-basiert) weiterentwickelt. Das System hat in allen Validierungsmaßnahmen seine Praxistauglichkeit unter Beweis stellen können (siehe Hauptteil des Berichts weiter unten).

Aktivitäten: Maßnahmen und Umsetzung

Aktivitäten / Maßnahmen im Darmstädter Teilprojekt sind die im Antrag spezifizierten Arbeitspakete. Entgegen der ursprünglichen Planung, in den ersten Monaten des Projektes nur an Konzepten zu arbeiten, haben wir sehr früh begonnen, erste Ideen prototypisch zu implementieren, um schnell zu ersten, vorläufigen Ergebnissen zu kommen und Annahmen und Strategien zur Machbarkeit/Umsetzbarkeit zu validieren. Darüber hinaus gab es nur untergeordnete Abweichungen vom Zeitplan; alle Arbeitspakete wurden ausreichend früh abgeschlossen, so dass alle Ziele erreicht werden konnten.

Die im Rahmen des Projekts entwickelte und erweiterte Software wurde umfassend durch Computerstudien bezüglich Antwortzeiten und Skalierbarkeit validiert, auch diese Arbeitspakete blieben vollständig im Zeitrahmen.

Das Darmstädter Teilprojekt verausgabte im Wesentlichen Personalmittel (inklusive studentischer Mitarbeiter). Im Projektverlauf hatte sich herausgestellt, dass erheblich größere Teile der Softwareentwicklungsarbeiten auf qualifizierte studentische Mitarbeiter übertragen werden können, als es ursprünglich möglich erschien. Dies betrifft das Arbeitspaket 4.3, insbesondere die Arbeit am Demonstrator und an der Bedienoberfläche. Dank dem Projekt *MOTIS* erfordern verschiedene einzelne Schritte in diesen beiden Teilpaketen keine tiefgehende wissenschaftliche Bearbeitung mehr (die wurde im Projekt *MOTIS* schon geleistet), sondern nur noch eine softwaretechnische Umsetzung, die zwar nicht über das Qualifikationsniveau studentischer Mitarbeiter hinausgeht, aber dennoch erheblichen Zeitaufwand erfordert. Abgesehen von dieser Verschiebung ist unsere ursprüngliche Kalkulation eingehalten worden.

Erfahrungen und Ergebnisse / Schlussfolgerungen und Perspektiven - im Sinne einer Optimierung

Ideen möglichst früh praktisch (in diesem Fall durch die Entwicklung prototypischer Software) zu validieren um die Konzeptionsphase zu unterstützen - ein Kerngedanke der agilen Softwareentwicklung -, hat sich als äußerst effektiv herausgestellt. Hierdurch konnten schnell Erfolge erzielt werden.

Tatsächlich waren alle wissenschaftlichen und studentischen Mitarbeiter schon in ihrem Bachelorstudium an der TU Darmstadt durch ein internes Pflichtprojekt intensiv theoretisch und vor allem praktisch in professioneller Herangehensweise an eine reale Projektsituation unter festen Terminvorgaben geschult worden (dieses Lehrformat hatte der Antragsteller Weihe 2002 entwickelt und ins Bachelorstudium eingeführt, und dafür hat er 2012 den Athene-Preis der Giersch-Stiftung erhalten). Das Projekt Per Pedes Routing, in dem alle Ziele wie gesagt zeitgerecht erreicht werden konnten, ist geradezu ein Musterbeispiel für die Sinnhaftigkeit einer solchen Lehrveranstaltung in einem Informatikstudiengang.

Im nun folgenden Hauptteil wird detailliert auf die entwickelten Konzepte und erzielten Ergebnisse eingegangen. Schlussfolgerungen und Ausblick schließen das Dokument ab.

Einleitung des Hauptteils

Die Forschungen, die im Projekt Per Pedes Routing durchgeführt wurden, lassen sich in zwei ineinandergreifende Komponenten unterteilen. Zum einen wurde ein speziell für mobilitätseingeschränkte Menschen personalisiertes Fußgänger-Routing entwickelt und in die Fahrplanauskunft MOTIS zur Berechnung des Weges im ersten und letzten Abschnitt der Route integriert; dies war die primäre Aufgabe der Technischen Universität Darmstadt. Zum anderen wurde für die Erfassung und Identifikation der für das Routing benötigten Infrastrukturdaten (insbesondere über Art und Lage von Hindernissen) ein crowdsourcing basierter Ansatz und die zugehörige IT-Unterstützung entwickelt; hierfür hat die Hochschule Fresenius einen "offline" Ansatz entwickelt und in intensiven Feldversuchen eingesetzt sowie die Ergebnisse evaluiert und umgesetzt. Die Technische Universität Darmstadt hat darüber hinaus noch einen "online" Ansatz in Form einer App beigetragen. Alle Softwarekomponenten, die an der TU Darmstadt entwickelt wurden, verwenden die öffentlich zugänglichen, crowdsourcing-basierten OpenStreetMap (OSM) Daten. Es hat sich herausgestellt, dass die Daten in OpenStreetMap inklusive Barriere Daten schon ausreichend vollständig sind, um unsere Softwarekomponenten auf dieser Basis zu entwickeln und zu evaluieren. Im Folgenden konzentriert sich dieser Bericht auf die Arbeiten an der TU Darmstadt.

Verwandte Arbeiten

Im Gegensatz zu vielen bisherigen wissenschaftlichen Veröffentlichungen, die sich stark auf Rollstuhlfahrer (z.B. [1] und [2]) konzentrieren, hat das Projekt Per Pedes Routing eine umfassendere Zielgruppe, die insbesondere ältere Menschen mit typischen altersbedingten Mobilitätseinschränkungen mit einschließt. Ohne jede Änderung an der Software ließen sich auch eingeschränkte Nutzer wie beispielsweise "zierliche Person mit schwerem Kinderwagen" oder die Vermeidung von Angstzonen für risikoaverse Nutzer (mit individuellem Grad von Aversion) integrieren.

Bezogen auf das grobe Konzept und die Systemarchitektur ähnelt Per Pedes Routing dem Projekt mPASS [5]. In wichtigen Details, insbesondere beim Routing (siehe unten) unterscheidet sich allerdings der gewählte Ansatz von [5]. Um das breit gefächerte Spektrum an Einschränkungen in allen Zielgruppen abzudecken, ist das Routing im Per Pedes Routing Projekt personalisiert (individualisiert). Um Hindernisse rechnerisch handzuhaben, verlässt sich Per Pedes Routing weder auf eine gewichtete Summe zur Verrechnung von Hindernissen mit Reisezeit (im Gegensatz z.B. zu [3] und [5]), noch allein auf einen totalen Ausschluss von Wegen mit Barrieren (wie in [1] und [2]). Stattdessen wurde für Per Pedes Routing ein (im Bereich des Fußgänger-Routings) neuartiger multikriterieller Ansatz entwickelt, der dafür sorgt, dass die Nutzer/innen aus einer Menge von verschiedenen Routen auswählen können, auf denen überwindbare Hürden die Bewertung der Route verschlechtern (siehe Abschnitt "Routing" unten). Praktisch läuft das darauf hinaus, dass die Nutzer nicht bei jeder Suchanfrage selbst die Entscheidung treffen müssen, sondern dass ein

Abgleich mit dem individuellen Nutzerprofil (s.u. bei Routing) vorgenommen wird, um dem Nutzer nur noch die für sein Profil potentiell optimalen Wege vorzuschlagen.

Bei der Datenerfassung wurde ähnlich wie in [5] ein Sensordaten-basierter Ansatz entwickelt und durch einen manuellen (Android App) basierten Ansatz ergänzt.

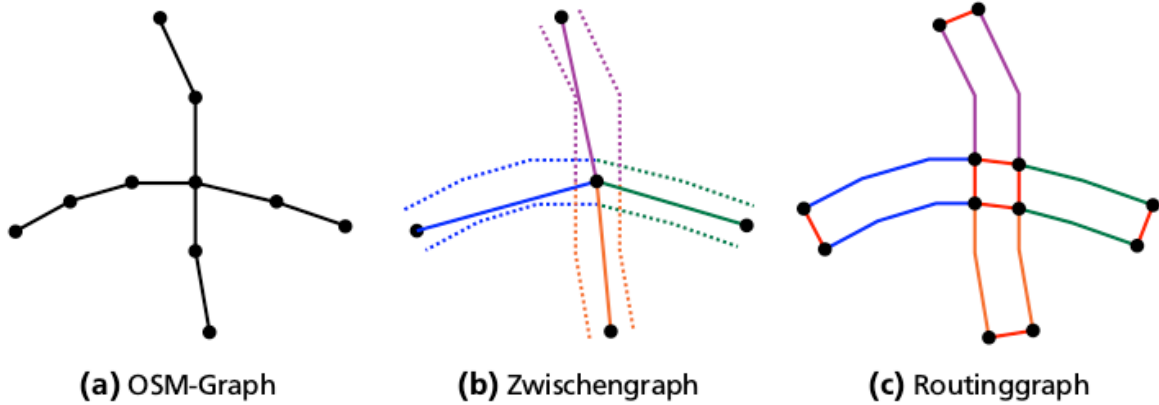
Im Laufe des Projektes sind unseres Wissens keine neuen relevanten wissenschaftlichen oder kommerziellen Ergebnisse andernorts erzielt worden.

Routing

Grundsätzlich ist bei vielen mobilitätseingeschränkten Menschen nicht davon auszugehen, dass Hindernisse unüberwindbar sind. Eine binäre Entscheidung "Hindernis zumutbar ja/nein" ist daher inadäquat, denn würden Hindernisse wie beispielsweise Treppen grundsätzlich ausgeschlossen, kann dies zu unnötig großen Umwegen führen: Ist beispielsweise der Umweg, um eine kurze Treppe zu umgehen, mehrere hundert Meter lang, sind die Nutzer/innen je nach Einschränkunggrad sicherlich an beiden Alternativen interessiert, um selbst entscheiden zu können. Ein kategorischer Ausschluss von Wegen, die bestimmte Hindernisse enthalten, ist folglich nicht zielführend.

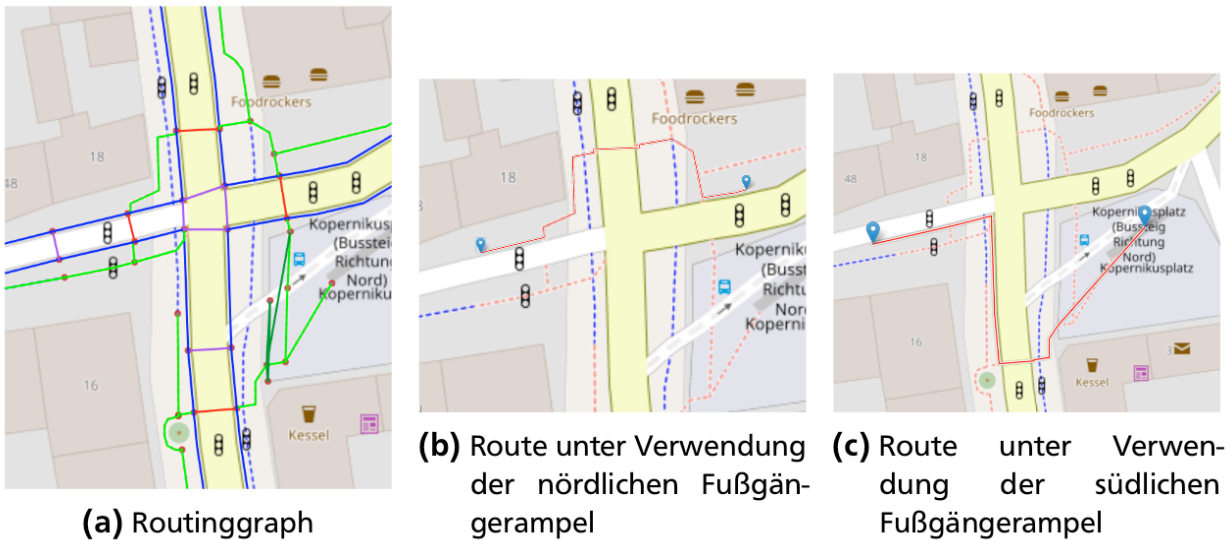
Das von uns entwickelte Routing findet optimale Wege bezogen auf die beiden Kriterien Laufzeit und Beschwerlichkeit. Da die Beschwerlichkeit eines Hindernisses und die Laufgeschwindigkeit von der Art und Schwere der Einschränkungen der Nutzer/innen abhängt, wurde das Routing personalisiert. Das heißt, das spezifische Profil der Einschränkungen des Menschen (Basisparameter wie Laufgeschwindigkeit sowie individuelle Einschränkungen) wird während der Berechnung der Wege berücksichtigt. Wie schon oben erwähnt, erlaubt unsere Software beliebige Nutzerprofile, nicht nur solche, die durch physische Einschränkungen bedingt sind.

Darüber hinaus berücksichtigt der Algorithmus beim Routing wichtige Detailinformationen wie bspw. unterschiedliche Straßenseiten (siehe Grafik 1), und er verwendet beim Wechsel der Straßenseite bevorzugt Fußgängerampeln und Zebrastreifen (siehe Grafik 2). Der Wechsel der Straßenseite ohne Fußgängerübergang ist ein weiteres Beispiel für die Allgemeinheit unseres Ansatzes im Vergleich zu anderen, denn ein solcher Wechsel kann einfach als weitere Art von überwindbarer Barriere, die für verschiedene Nutzerprofile unterschiedlich beschwerlich ist, integriert werden.



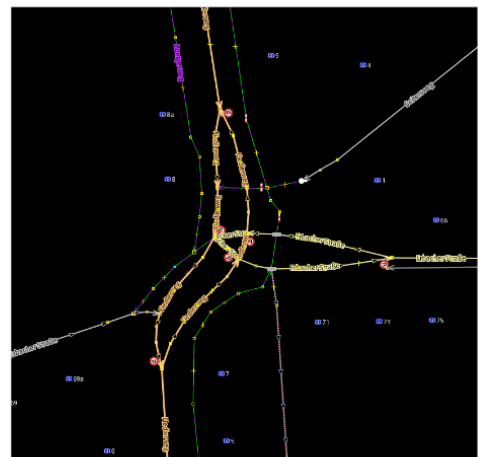
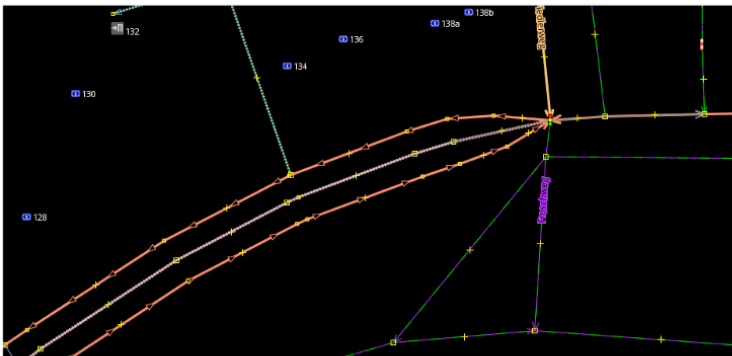
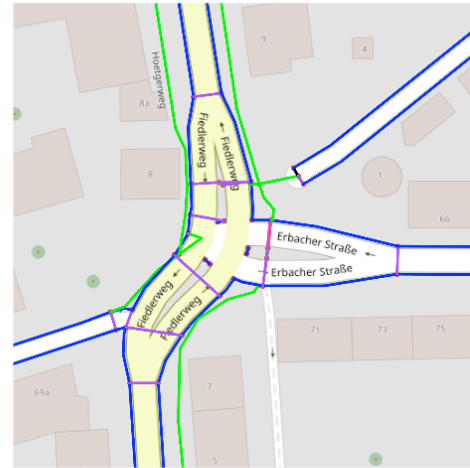
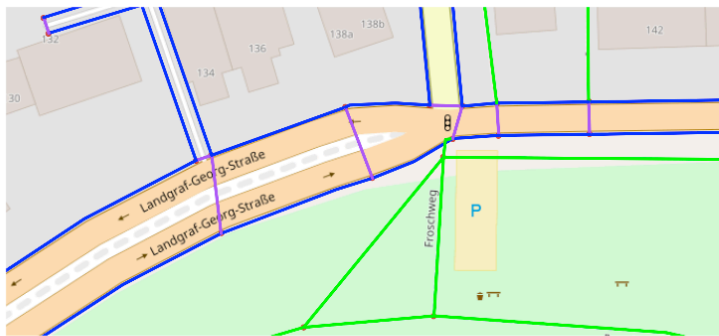
Gr

Grafik 1: Berücksichtigung verschiedener Straßenseiten im Rahmen der grundlegenden Transformation der OpenStreetMap Daten in den Routing Graph



Grafik 2: Straßenüberquerung; lila Straßenquerungen sind Querungen ohne Querungsanlage (Ampel/Zebrastrreifen), rote Straßenquerungen sind Querungen mit Querungsanlage; grüne Wege sind Fußwege aus OpenStreetMap; blaue Wege sind Gehwege, die von unserer Datentransformation aus reinen Straßendaten (ohne Gehwege) generiert wurden

Straßen, bei denen die beiden Fahrbahnen als zwei unabhängige Straßen in OpenStreetMap abgebildet sind, werden speziell behandelt, um zu verhindern, dass zwischen den beiden Straßen ein Gehweg im Routing-Graph erzeugt wird (siehe Grafik 3).



(a) Straße mit zwei getrennten Fahrbahnen (orange) sowie Busspur in der Mitte (b) Im Kreuzungsbereich separat eingezeichnete Fahrbahnen

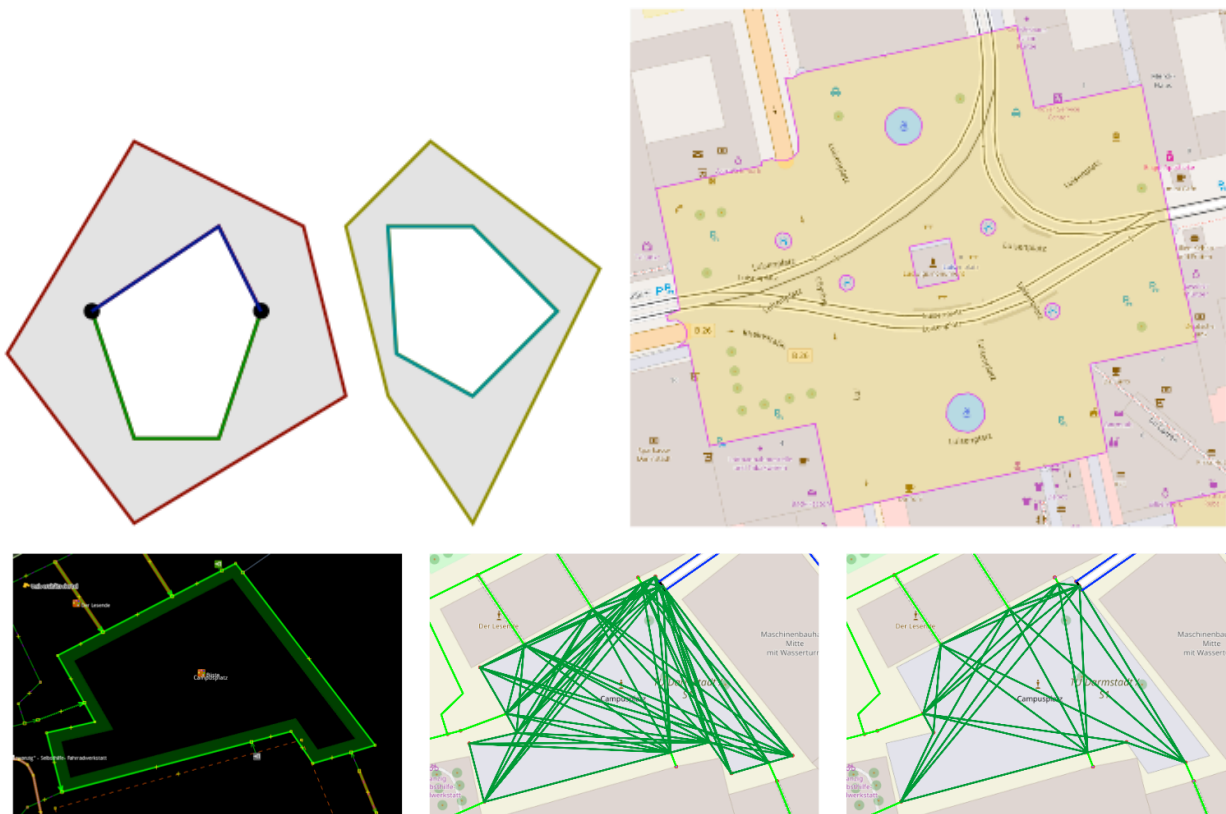
Grafik 3: Vereinigung mehrerer Fahrbahnen einer Straße in verschiedene Richtungen

Für Plätze und Flächen wird ein optimierter Routing-Graph berechnet (siehe Grafik 4).

Der Algorithmus bietet zudem die Möglichkeit, ein Höhenprofil (z.B. aus Lasermessungen aus der Luft) zu berücksichtigen. Höhenunterschiede werden in diesem Fall in die Beschwerlichkeit des Weges eingerechnet.

Auch dies ist ein Beispiel für die Allgemeinheit des von uns gewählten Ansatzes.

Um zu allen umliegenden Haltestellen gleichzeitig zu routen, wurde ein spezieller Anfragetyp implementiert. Dies ist eine Vorbereitung für die Einbindung in Reiseketten mit öffentlichen Verkehren, denn häufig kommen mehrere Haltestellen in Fußwegdistanz vom Startpunkt für die Weiterfahrt zum Ziel infrage.



(a) OpenStreetMap-Daten **(b)** Vollständiger Visibility Graph **(c)** Reduzierter Visibility Graph

Grafik 4: oben Beispiele für Plätze, dargestellt als Multipolygon (innere Polygone sind nicht begehbare Flächen, z.B. Gebäude auf dem Platz; rechts der Luisenplatz in Darmstadt mit der Ludwigssäule und zwei Brunnen als nicht begehbaren Flächen). Unten: Generierter Routinggraph vor und nach Optimierung (Reduktion der Kantenzahl)

Validierung

Algorithmen / Software für ein Optimierungsproblem sind grundsätzlich nach zwei Kriterien zu evaluieren: erstens nach der Güte der vom Algorithmus berechneten Ergebnisse - also wie weit weg die Ergebnisse vom Optimum gemäß individuellem Nutzerprofil entfernt sind unter der Voraussetzung, dass die Eingabedaten korrekt sind - und zweitens die Antwortzeiten. Diese beiden Kriterien stehen in einem Widerspruch zueinander: Je bessere Ergebnisse man haben möchte, umso länger werden in der Regel die Antwortzeiten sein. Bei unserem algorithmischen Ansatz kann man prinzipiell verschiedene Kompromisse zwischen diesen beiden Kriterien durch Voreinstellungen schließen. Bei den validierenden Computerstudien waren die Voreinstellungen im Algorithmus allerdings so getroffen, dass die Ergebnisse garantiert optimal sind. Die Frage, die die Computerstudien beantworten sollten, war daher, wie die Antwortzeiten auch unter dieser maximal strengen Vorgabe sein würden. Sollten die Antwortzeiten mit diesen Einstellungen schon ausreichend gut sein, erübrigen sich Computerstudien mit suboptimalen Einstellungen mangels Bedarfs.

	∅	Q(50)	Q(80)	Q(90)	Q(99)
Fußgängerrouting	425 ms	322 ms	611 ms	884 ms	1697 ms
ÖV-Routing	462 ms	451 ms	680 ms	831 ms	1111 ms
Gesamtdauer	887 ms	773 ms	1291 ms	1715 ms	2808 ms

Grafik 5 (a): Rechenzeiten für das intermodale Routing unterteilt in Rechenzeiten für das Fußgänger-Routing und das ÖV-Routing

Einer der Meilensteine für das Projektende gemäß Antrag war eine umfassende Studie mit Daten der Deutschen Bahn. Hier kommt das Fußgänger-Routing für die Fußwege von der Start-Adresse zur ersten ÖV-Haltestelle/Station und auf dem Weg von der letzten ÖV-Haltestelle/Station zur Zieladresse zum Einsatz, was ja der Fokus von Per Pedes Routing ist. Die Antwortzeiten sind in Grafik 5 (a) zu sehen. Die Evaluation wurde mit allen Fahrten des öffentlichen Verkehrs (Fernzüge, Nahverkehrszüge, Busse, Straßenbahnen, usw.) durchgeführt. Die Auswertung schließt ganz Deutschland ein. Für intermodale Anfragen benötigt das System im Schnitt weniger als eine Sekunde (887 Millisekunden). Selbst aufwendige Anfragen können in weniger als drei Sekunden (99% Quantil 2808ms) beantwortet werden.

Damit ist die Praxistauglichkeit prinzipiell nachgewiesen.

Ein wesentlicher Aspekt des Kriteriums Antwortzeiten ist die Skalierbarkeit, also das Verhalten der Antwortzeiten bei wachsenden Datenmengen. Daher wurde das im Projekt Per Pedes Routing neu entwickelte Fußgänger-Routing speziell auch auf seine Skalierbarkeit hin analysiert. Die Ergebnisse sind in Grafik 5 (b) dargestellt. Hier ist deutlich erkennbar, dass auch ein OpenStreetMap Datensatz, der ganz Deutschland umfasst, ohne Probleme im Routing verwendet werden kann.

Damit ist auch die Skalierbarkeit nachgewiesen.

	Frankfurt	Hessen	Deutschland
Größe der OSM-Eingabedatei	11,76 MiB	216,85 MiB	3,08 GiB
Größe der Ausgabedatei	35,55 MiB	611,08 MiB	7,86 GiB
OSM-Graph			
Knoten	152.580	4.567.335	55.949.081
Kanten	175.582	4.935.434	60.034.344
Flächen	582	3.374	35.539
Fußwege	76.340	3.087.221	33.510.863
Straßen	66.097	1.698.912	24.659.315
Gleise	25.093	116.585	1.501.742
Area-Kanten	8.052	32.716	362.424
Zwischengraph			
Knoten	61.648	1.118.815	14.328.883
Kanten	90.565	1.519.361	18.809.877
Fußwege	46.075	815.274	8.795.783
Straßen	37.171	676.264	9.643.309
Gleise	7.319	27.823	370.785
Routinggraph			
Knoten	107.454	2.010.688	27.204.790
Kanten	221.549	4.088.066	53.690.513
Fußwege	46.075	815.274	8.795.783
Gehwege	71.330	1.331.957	19.039.292
Aufzüge	198	518	9.098
Übergänge	101.280	1.864.759	24.877.814
Übergänge über eine Fahrbahn	97.797	1.850.578	24.707.231
Übergänge über mehrere Fahrbahnen	3.483	14.181	170.583
Unmarkierte Übergänge	94.844	1.837.039	24.609.026
Übergänge mit Zebrastreifen	2.068	9.199	77.341
Übergänge mit Verkehrsinsel	162	2.950	31.196
Übergänge mit Fußgängerampel	4.206	15.571	160.251
Performance			
	Frankfurt	Hessen	Deutschland
Gesamtdauer (ohne Serialisierung)	1,67 s	28,73 s	6 m 59 s
Serialisierung	240 ms	6,43 s	5 m 22 s*
Aufbau des OSM-Graphen	764 ms	13,21 s	3 m 16 s
Aufbau des Zwischengraphen	182 ms	2,87 s	40,80 s
Aufbau des Routinggraphen	727 ms	12,65 s	3 m 02 s
Extraktion der OSM-Daten	666 ms	9,99 s	2 m 32 s
Relationen	46 ms	500 ms	6,84 s
Knoten und Kanten	280 ms	6,92 s	1 m 43 s
Flächen (inkl. kürzeste Wege)	299 ms	1,26 s	22,47 s
Berechnung der Höhenprofile	98 ms	3,22 s	43,86 s
Erkennung mehrerer Fahrbahnen	93 ms	1,08 s	16,10 s
Gehwegpfade und Kantenreduktion	66 ms	1,35 s	17,92 s
Behandlung von Kreuzungen	282 ms	6,23 s	1 m 28 s
Übergänge über mehrere Fahrbahnen	69 ms	520 ms	6,53 s
Kantenerzeugung	96 ms	2,41 s	32,66 s
Umwege mit markierten Übergängen	277 ms	3,47 s	54,40 s

Grafik 5 (b): Statistiken zur Skalierbarkeit des entwickelten Routing-Algorithmus, hier: Vorbereitung (Preprocessing) der Daten

Datenerfassung

Dieser Abschnitt beschreibt nur den "online" Ansatz, der direkt auf der App ausgeführt wird. Der "offline" Ansatz wird im Bericht der Hochschule Fresenius beschrieben, da dies, wie oben erwähnt, in den Aufgabenbereich der Hochschule Fresenius fiel.

Window	ACC				FPR				DataType
	RF	J48	SMO	BN	RF	J48	SMO	BN	
1	95.80	94.43	84.21	85.06	0.05				Accelerometer
2	98.23	97.49	87.69	92.06	0.02	0.03	0.09	0.09	
3	98.55	97.82	91.81	93.48	0.02	0.02	0.06	0.08	
4	98.93	98.11	93.62	94.78	0.02	0.02	0.04	0.06	
5	99.29	98.53	95.22	95.36	0.01				
1	80.23	78.16	72.12	70.69	0.22				Gyroscope
2	86.66	84.33	76.55	75.12	0.14	0.13	0.13	0.25	
3	89.95	87.34	78.77	76.61	0.10	0.10	0.10	0.26	
4	91.85	89.02	79.67	76.22	0.08	0.09	0.09	0.27	
5	92.37	90.00	79.43	77.39	0.08				
1	96.11	94.18	87.33	82.05	0.05				Combined
2	98.19	97.12	91.38	87.74	0.02	0.03	0.06	0.17	
3	98.87	97.90	94.38	89.92	0.01	0.02	0.03	0.15	
1	60.01				0.45				Pressure
2	66.17				0.37				
3	73.42	77.47	75.02	77.03	0.30				
4	78.19	81.31	80.41	80.56	0.25	0.21	0.16	0.22	
5	83.91	85.39	85.14	85.42	0.19	0.16	0.12	0.15	
6	87.19	88.26	88.56	88.93	0.14	0.13	0.09	0.12	
7	89.59				0.10				
8	92.37				0.09				
1	95.74	94.70	87.63	84.49	0.05				All
2	97.92	96.88	91.40	90.49	0.03	0.03	0.06	0.15	
3	98.84	97.96	94.89	93.21	0.02				
4	98.67	97.67	96.32	94.33	0.02				
5	99.08	98.42	97.29	95.88	0.01				

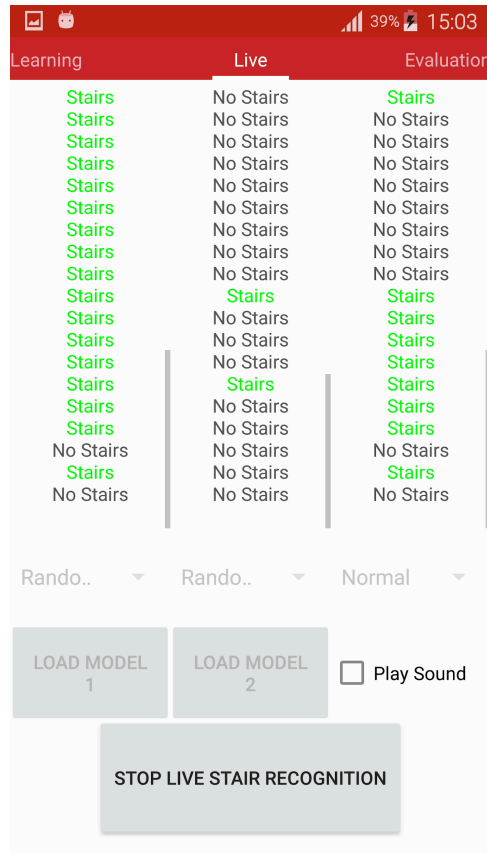
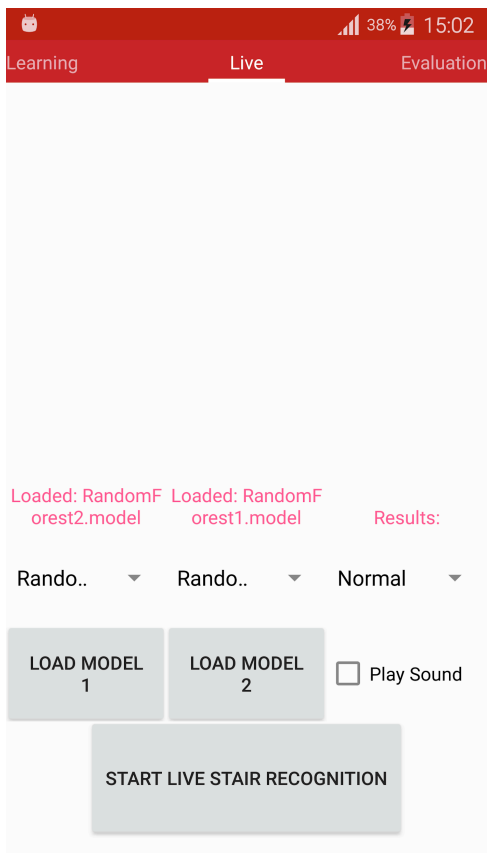
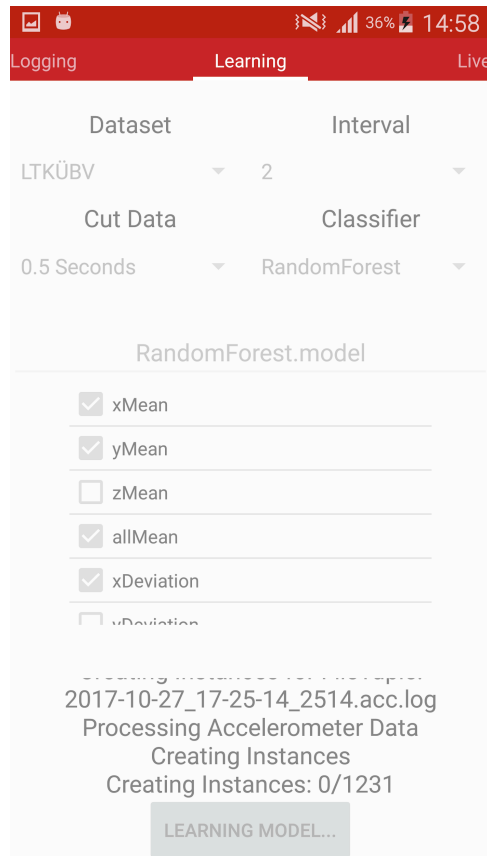
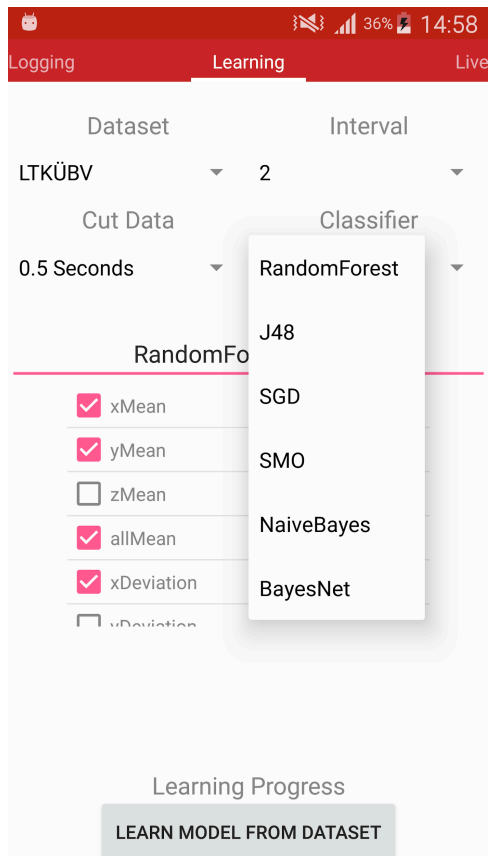
Grafik 5 (d): Erkennungsraten (ACC) und Rate der Fehler zweiter Art (FPR) der verschiedenen Lern-Algorithmien (Random Forrest, J48 Decision Tree, Sequential Minimal Optimization, Bayesian Network) unter Verwendung verschiedener Analyse-Zeit-Intervalle ("Window") in Sekunden und verschiedener Sensordaten (Accelerometer, Gyroskop, Accelerometer und Gyroskop, Barometer, alle kombiniert)

Im Bereich der Datenerfassung wurde an der TU Darmstadt eine mobile Applikation ("App") für die Android Plattform entwickelt (siehe hierzu Grafik 5 (c)). Diese ermöglicht es, mit Hilfe von Sensoren, die auf handelsüblichen Smartphones zur Verfügung stehen (beispielsweise Beschleunigungssensor und Gyroskop), automatisiert zu erkennen, ob die Nutzerin/der Nutzer, die/der das Gerät mit sich führt, gerade eine Treppe benutzt oder nicht. Der entwickelte Ansatz ist dahingehend generisch, dass die App zum Lernen von jeder Art von Bewegungsmustern verwendet werden kann. Sie lässt sich folglich auch auf andere Hindernisse anwenden. Im Rahmen dieses Projekts haben wir uns auf Treppen als Beispielmaterial konzentriert. Unser System ist für verschiedene Erweiterungen vorbereitet, zum Beispiel die erkannte Treppe vollautomatisiert

mit GPS Daten zu verknüpfen und an einen Server zu melden, der die Treppe mit ihrer Lokalisation ebenso automatisch in einen Datenkorpus einpflegt. In der Praxis würde das sicherlich darauf hinauslaufen, dass der Server nur dann neue Daten automatisiert einpflegt, wenn er von mehreren Nutzer/innen weitgehend unwidersprüchliche Informationen erhält, so dass er sicher davon ausgehen kann, dass sich an dieser übereinstimmend genannten Position tatsächlich eine Treppe befindet.

Wie in Grafik 5 (d) dargestellt, liegt die Erkennungsrate unter kontrollierten Bedingungen (dieselbe Person, kontrollierte Smartphone Position) weit über 90 Prozent. Die beste Klassifikation der Bewegung (mit einer Genauigkeit von über 99 Prozent) basiert auf dem im Bereich des maschinellen Lernens anzuesiedelnden Random Forest Klassifikationsverfahren.

Die Datenverarbeitung (Klassifikation der erfassten Sensordaten) findet direkt auf dem Endgerät statt, so dass anstelle der umfangreichen Sensordaten ausschließlich das Endergebnis (Position der erkannten Treppe) an den Server übertragen werden muss. Das hat den Vorteil, dass zur Übertragung der Daten weniger Energie und Bandbreite benötigt wird. Ergänzend zu der automatisierten Erfassung von Treppen wurde eine weitere Android App entwickelt (Details im Abschnitt "Demonstrator"), die es erlaubt, den für das Routing genutzten OpenStreetMap Datensatz mit zusätzlichen Wegen zu versehen. Diese App erlaubt auch die manuelle Attributierung von Treppen (z.B. Informationen über Geländer).



Grafik 5 (c): An der TU Darmstadt entwickelte mobile Android App zum Lernen von Bewegungsmustern mittels auswählbarer Machine Learning Algorithmen, die auf frei auswählbare Smartphone Sensoren angewendet werden

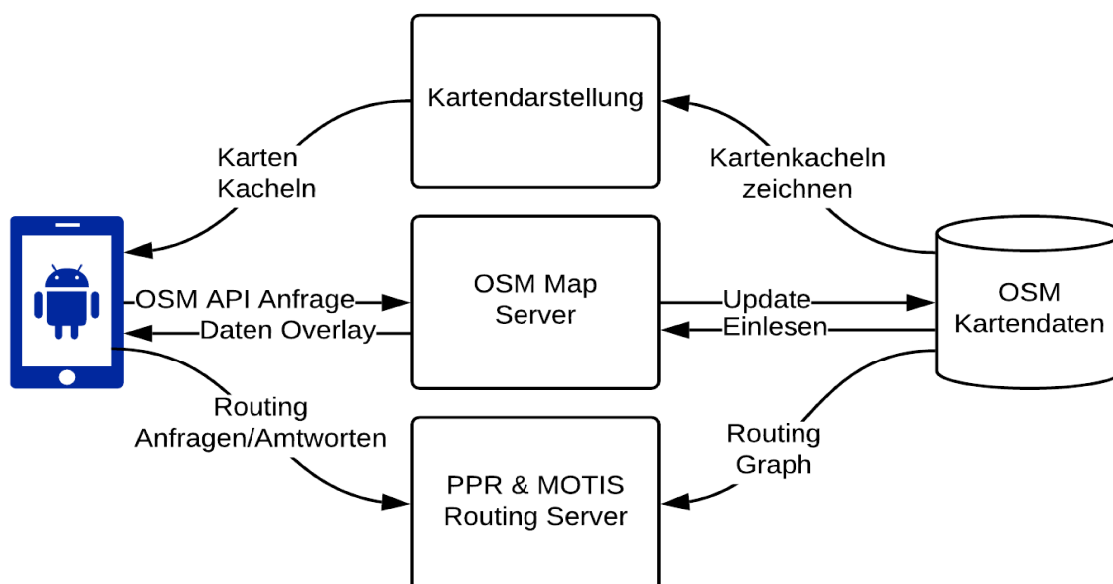
Demonstrator

In Abschnitt 4.3 des Antrags wurde ein Demonstrator in Aussicht gestellt.

Der entwickelte Demonstrator besteht aus verschiedenen Komponenten, die jeweils Aufgaben im Bereich der Datenerfassung und des Routings übernehmen. Als Datenbasis werden die Kartendaten aus dem OpenStreetMap Projekt verwendet. Zusätzliche Daten können mit einer speziell im Per Pedes Routing Projekt entwickelten Android App erfasst werden. Zu den erfassbaren Daten gehören ganz neue Wege, aber auch Hindernisse wie Treppen mit allen für das Routing nötigen Attributen: Richtungsinformation, Informationen über Geländer sowie die Anzahl der Stufen der Treppe.

Der gesamte Prozess von der Erfassung der Daten bis hin zum Routing kann in den Grafiken 6, 7 und 8 nachvollzogen werden:

- In Grafik 8 ist der ursprüngliche Zustand und das Routing Ergebnis basierend auf diesem Zustand dargestellt; da auf dem Weg keine Hindernisse vorliegen, wird genau ein Ergebnis (der kürzeste Weg) ausgegeben.
- Grafik 6 und 7 zeigen, wie mit der eigens entwickelten Android App Wege und Treppen auf Wegen in den OpenStreetMap Datensatz integriert werden können.
- Der untere Teil von Grafik 8 zeigt jetzt den neuen Zustand und das dazu passende Routing-Ergebnis. Jetzt gibt es zwei Wege, die vom Router ausgegeben werden. Beide sind auf ihre Weise optimal: Der alte Weg (in der Grafik blau dargestellt) enthält - weiterhin - keine Hindernisse. Der durch die Änderung der Daten jetzt zusätzlich ermöglichte Weg (in der Grafik rot dargestellt) ist zwar kürzer, enthält aber eine Treppe.



Grafik 5 (d): Software Architektur des Demonstrators

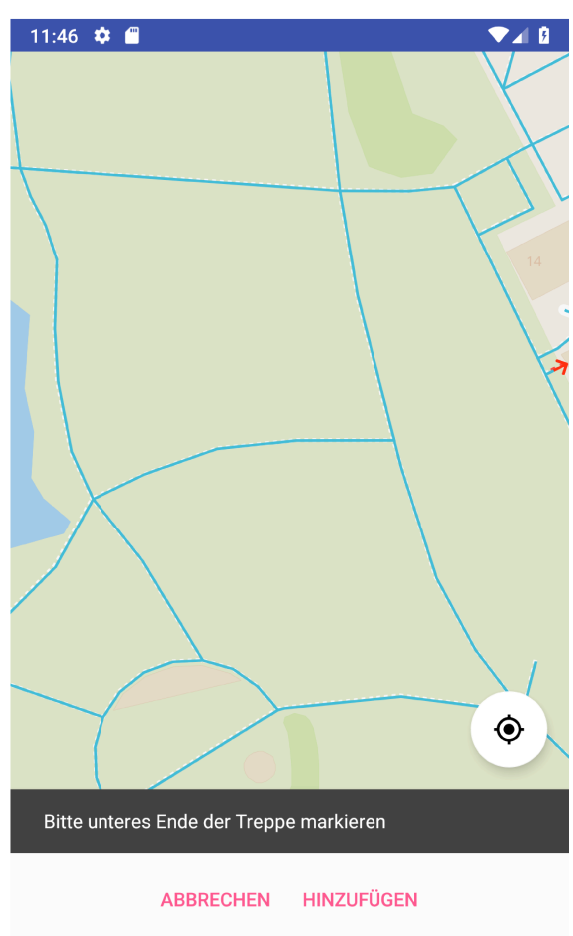
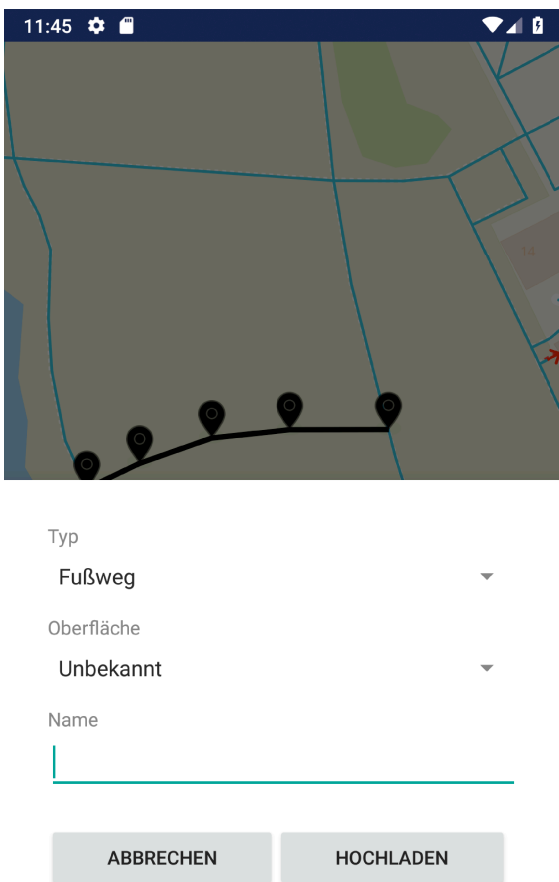
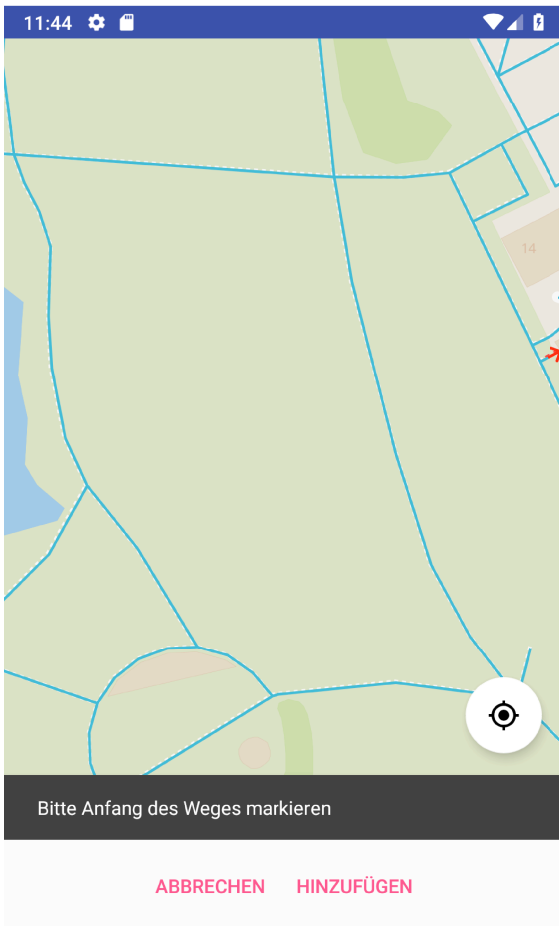
Grafik 5 (d) zeigt die technische Architektur der Softwarekomponenten des Demonstrators. Die Kommunikation zwischen allen Komponenten basiert so weit wie möglich auf existierenden Standards:

- Die zur Kartendarstellung der (ggf. durch neu erfasste Wege und Hindernisse erweiterten) OpenStreetMap Daten werden zwischen dem Server, der die Kartendarstellung bereitstellt und der Android Applikation mittels Vector Tiles¹ ausgetauscht.
- Das Overlay (blau markierte Wege in Grafik 6 und 7) wird in Form von OSM XML² vom OSM Map Server für die Android Applikation bereitgestellt.
- Mit der App erzeugte Änderungen am Datensatz (z.B. neue Wege oder Treppen) werden mittels OsmChange³ an den Server übermittelt und von diesem in den OpenStreetMap Datensatz integriert. Da dieser als Quelle für die Kartendarstellung und das Routing dient, bekommen alle Nutzer/innen eine aktualisierte und konsistente Sicht auf den Datenbestand.

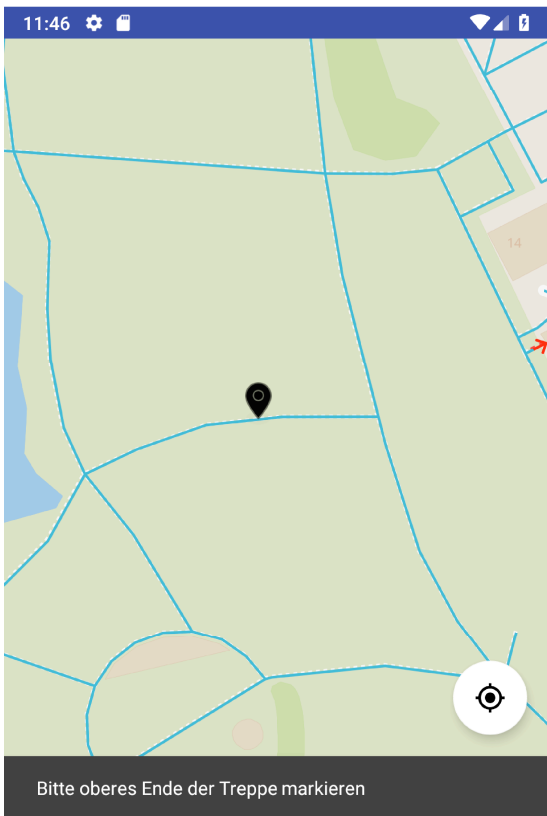
1 <https://docs.mapbox.com/vector-tiles/specification/>

2 https://wiki.openstreetmap.org/wiki/OSM_XML

3 <https://wiki.openstreetmap.org/wiki/OsmChange>



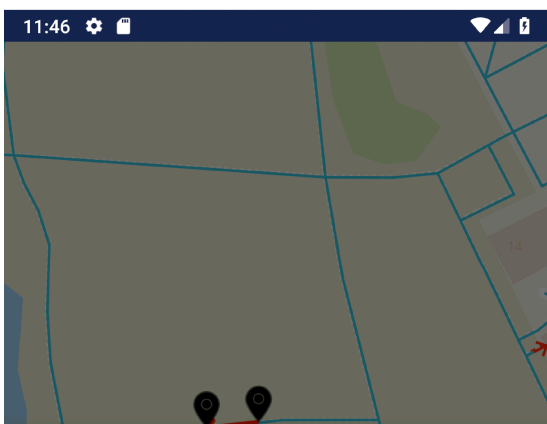
Grafik 6: Erfassung eines Fußweges mittels entwickelter Android Applikation



ABBRECHEN HINZUFÜGEN



ABBRECHEN HINZUFÜGEN



Anzahl der Stufen

Name

Handlauf

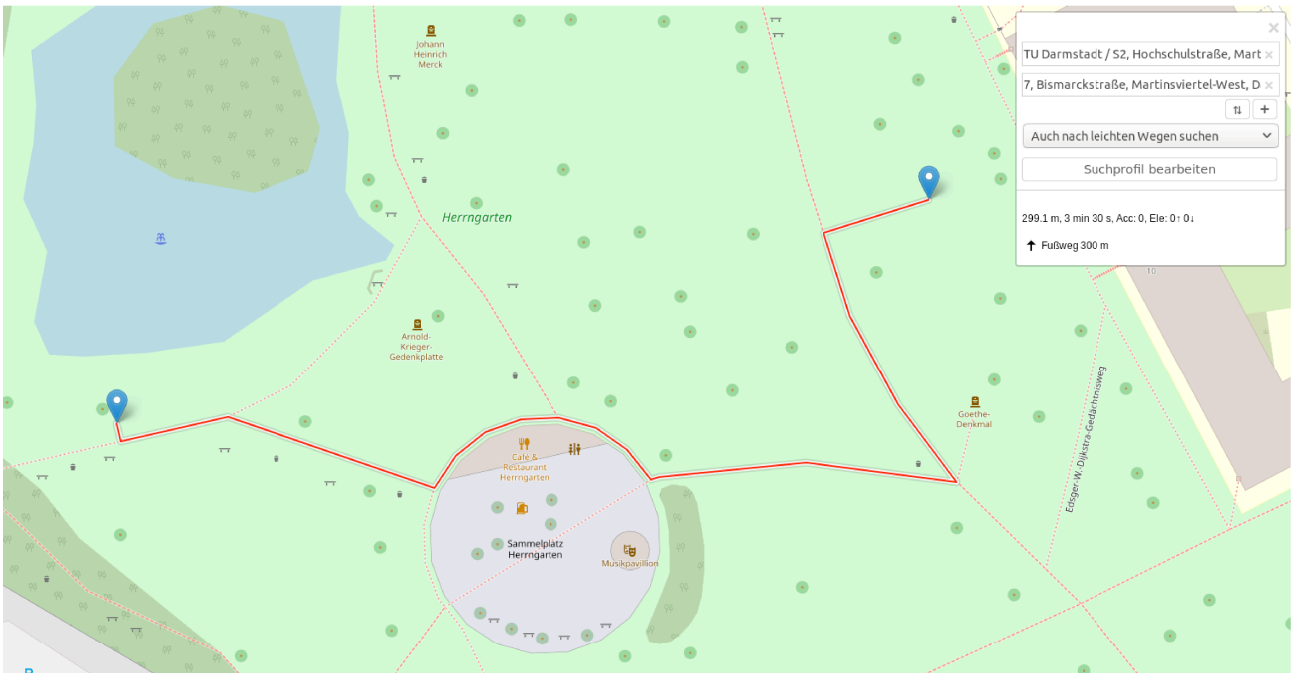
Rampe

ABBRECHEN

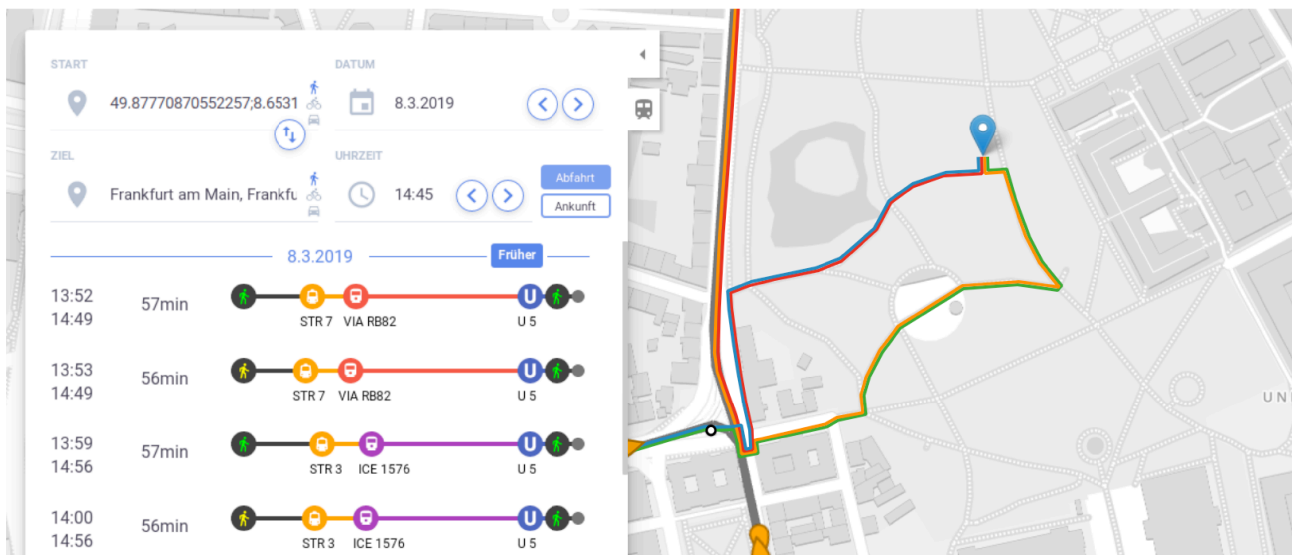
HOCHLADEN



Grafik 7: Erfassung von Treppen mittels entwickelter Android Applikation



Grafik 8: Entwickelte Routing Oberfläche ohne (oben) und mit neu erfasstem Weg (unten)



Grafik 9: Routing Web Oberfläche mit einem Beispiel für Wegealternativen. Gezeigt wird der Startpunkt in Darmstadt-Innenstadt (blauer Marker) sowie die in diesem Fall einzige infrage kommende Haltestelle (kleiner weißer Kreis mit schwarzem Rand) für die Weiterfahrt per ÖV nach Frankfurt am Main. Von dieser Haltestelle aus gibt es zwei optimale Alternativen per ÖV zum Ziel: eine Richtung Norden (rot und orange) und eine Richtung Westen (blau und grün). Die Haltestelle kann aus dem Park entweder über den kurzen Weg mit Treppe (blau und rot) oder über den etwas längeren Weg ohne Treppe (grün und orange) erreicht werden. Durch die beiden optimalen Pfade im ÖV ab der Haltestelle und die beiden optimalen Pfade zu Fuß bis zur Haltestelle ergeben sich insgesamt vier optimale Alternativen (blau, rot, orange und grün).

Das von uns entwickelte intermodale Auskunftssystem MOTIS wurde um das Fußgänger-Routing aus dem Per Pedes Routing Projekt erweitert. MOTIS erlaubt Tür-zu-Tür Verbindungen mit beliebigen Verkehrsangeboten. Die erste und letzte Etappe einer solchen Verbindung sind in der Regel Fußwege (in MOTIS alternativ auch Taxi, Fahrrad o.a. möglich). Dank Per Pedes Routing ist die Suche speziell bei Fußwegen in MOTIS nunmehr barriere sensitiv, das heißt, die Suche gezielt nach belastungsarmen und dennoch kurzen Wegen zu/von Haltestellen und Bahnhöfen (z.B. für ältere Reisende) ist jetzt möglich. Grafik 9 zeigt das oben erläuterte Beispiel - ein zusätzlicher Fußweg, der aber eine Treppe enthält - in Verbindung mit dem öffentlichen Verkehr: Hier wird eine Verbindung gefunden, die zwar eine Minute länger benötigt, dafür im Gegensatz zur schnellsten Verbindung aber keine Treppe enthält.

Zusammenfassung und Ausblick (voraussichtlicher Nutzen, Verwertbarkeit)

Es wurde von der Datenerfassung über die Datenhaltung bis zum Routing ein vollständiges System zum Berechnen von optimalen Routen für mobilitätseingeschränkte Menschen entwickelt. Mit Computerstudien konnte nachgewiesen werden, dass die bestmöglichen Routing-Vorschläge mit ausreichend geringen Antwortzeiten berechnet werden können.

Das Projekt Per Pedes Routing war auf Out-Door-Fußwege im städtischen Bereich fokussiert. Dies bietet für sich genommen sicherlich großes Potential und gute Anknüpfungspunkte für weitere Kooperationen.

Die konzeptuellen Ergebnisse von Per Pedes Routing und die auf dieser Basis entwickelten Softwarekomponenten waren zudem von Anfang an auf Verallgemeinerung und auf die Anwendung auf andere Szenarien angelegt. Das konkret von uns als nächstes ins Auge gefasste Anwendungsszenario ist In-Door-Routing in Bahnhöfen und die Verwendung der berechneten Wegezeiten als Umstiegszeiten zwischen verschiedenen Verkehrsmitteln des öffentlichen Verkehrs. Die Übertragung auf dieses Szenario wird auf der Basis von Per Pedes Routing möglich sein, aber aufgrund der engen Verflechtung mit dem Routing-Algorithmus für den öffentlichen Verkehr wird es voraussichtlich noch Forschungsbedarf in punkto Antwortzeiten geben. Das von unserer Darmstädter Arbeitsgruppe kontinuierlich weiterentwickelte Fahrplanauskunftssystem MOTIS würde dann die Möglichkeit bieten, die Fußwegesuchen out-door (Per Pedes Routing) und in-door (zukünftig) mit den schon in MOTIS integrierten öffentlichen Verkehrsangeboten und privaten Verkehrsmitteln zu beliebigen Reiseketten zu kombinieren. Damit wäre bedarfsgerechte Reiseplanung für Nutzer mit beliebigen Einschränkungen möglich.

Die im Projekt Per Pedes Routing an der TU Darmstadt erstellten Softwarekomponenten sind unabhängig von einem konkreten Auskunftssystem entwickelt und implementiert worden und lassen sich daher prinzipiell auch in andere, ähnliche Vorhaben integrieren.

Bibliographie

1. Kasemsuppakorn, P., & Karimi, H. A. (2009). *Personalised routing for wheelchair navigation*. Journal of Location Based Services, 3(1), 24-54.
2. Menkens, C., Sussmann, J., Al-Ali, M., Breitsameter, E., Frtunik, J., Nendel, T., & Schneiderbauer, T. (2011, April). *EasyWheel-A mobile social navigation and support system for wheelchair users*. In 2011 Eighth International Conference on Information Technology: New Generations (pp. 859-866). IEEE.
3. Völkel, T., & Weber, G. (2008, October). *RouteCheckr: personalized multicriteria routing for mobility impaired pedestrians*. In Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility (pp. 185-192). ACM.
4. Holone, H., Misund, G., & Holmstedt, H. (2007, September). *Users are doing it for themselves: Pedestrian navigation with user generated content*. In The 2007 International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST 2007) (pp. 91-99). IEEE.
5. Prandi, C., Salomoni, P., & Mirri, S. (2014, January). *mPASS: integrating people sensing and crowdsourcing to map urban accessibility*. In 2014 IEEE 11th Consumer Communications and Networking Conference (CCNC) (pp. 591-595). IEEE.