

# Offline-MapMatching – ein QGIS-Plug-in zum Abgleich einer Trajektorie mit einem Wegenetz

## *Offline-MapMatching – A QGIS Plugin for Matching a Trajectory with a Network*

Christoph Jung

jagodki.cj@gmail.com

**Zusammenfassung:** Map Matching bezeichnet das Problem des Abgleichs gemessener Koordinaten mit Informationen einer Karte, z. B. der Abgleich einer punktförmigen Trajektorie mit einem linienförmigen Wegenetz. Eine rein räumliche Analyse mit einem GIS erbringt in vielen Fällen keine zufriedenstellenden Resultate. Das Plug-in Offline-MapMatching stellt die erste Erweiterung für QGIS dar, um mittels stochastischer Methoden ein Map Matching zu realisieren. Diese Arbeit beschreibt den Prozessablauf, die mathematischen Grundlagen und die Parameter des Plug-ins.

**Schlüsselwörter:** Map Matching, QGIS-Plug-in, Hidden-Markov-Model, Viterbi-Algorithmus

**Abstract:** *Map Matching is the problem of comparing measured coordinates with the information of a map, e. g. the comparison of a point-shaped trajectory with a line-shaped network. A purely spatial analysis with a GIS does not deliver satisfactory results in many cases. The plugin Offline-MapMatching is the first extension for QGIS to realise a map matching with stochastic methods. This paper describes the process flow, the mathematical principles and the parameters of the plugin.*

**Keywords:** *Map matching, QGIS plugin, Hidden-Markov-Model, Viterbi algorithm*

## 1 Motivation und Stand der Technik

Der Abgleich gemessener Geodaten mit vorhandenem Kartenmaterial (Map Matching) ist ein häufig anzutreffendes Problem. Ein Anwendungsfall ist der Abgleich einer Trajektorie (der zurückgelegte Pfad eines bewegten Objektes im Raum repräsentiert durch die Abfolge einzelner gemessener Positionen (Andrienko et al., 2008, p. 18)) mit einem Wegenetz, wie er z. B. von Navigationssoftware beherrscht werden muss. Eine Lösung mit gängigen Geographischen Informationssystemen (GIS) aus dem Open-Source-Bereich ist jedoch nur unzureichend möglich. Um dieses Problem zu lösen, wurde das QGIS-Plug-in Offline-MapMatching entwickelt. Es handelt sich um das erste Plug-in für QGIS, um ein solches Verfahren zu bewerkstelligen. In dieser Arbeit wird auf die zugrunde liegenden Berechnungen und die Bedienung eingegangen.

Das Map Matching wird unterschieden in Online und Offline Map Matching. Die Unterscheidung bezieht sich auf den Zeitpunkt, wann das Verfahren stattfindet. Während das Online Map Matching kontinuierlich während der Datenaufnahme ausgeführt wird, findet das Offline Map Matching erst nach der Aufnahme der kompletten Trajektorie Anwendung (Budig, 2012). Genutzt wird das Offline Map Matching, um Bewegungsdaten auf Basis eines Wegenetzes zu analysieren, d. h. das Wegenetz stellt das Bezugssystem für die Trajektorien dar. Im Bereich der Verkehrsplanung können damit Straßenabschnitte identifiziert werden, welche stärker oder schwächer frequentiert sind. Für Befahrungskampagnen im Bereich Mobile

Mapping können durch das Map Matching die Straßenabschnitte ermittelt werden, welche schon befahren wurden und welche noch offen sind. Außerdem wird es bei der Zustandserfassung und -bewertung von Straßen eingesetzt, um Messdaten den entsprechenden Straßenabschnitten zuzuordnen.

Unterschiedliche Verfahren zum Bewerkstelligen des Map Matchings wurden in der Vergangenheit entwickelt und getestet, wobei sich stochastische Verfahren unter Zuhilfenahme eines Hidden Markov Models (HMM) in vielen Arbeiten als auch in der Industrie bewährt haben (Mattheis, 2015). Ziel solcher Ansätze ist es, jenen Weg im Wegenetz zu finden, der mit höchster Wahrscheinlichkeit die gegebene Trajektorie darstellt.

Einen oft zitierten Ansatz zur Lösung des Offline Map Matchings mittels HMM findet sich bei Lou et al. (2009). Dabei stellen Messungenauigkeiten der Trajektorie eine allgemeine Herausforderung dar, welchen mittels Definition weiterer stochastischer Parameter begegnet werden kann (Newson & Krumm, 2009). Eine Übersicht über verschiedene Ansätze des Offline Map Matchings und der verwendeten Parameter liefert Budig (2012).

## 2 Funktionsweise

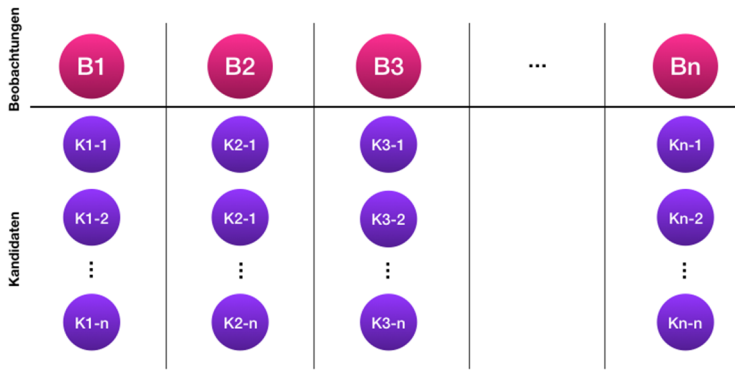
### 2.1 Allgemeiner Prozessablauf

Kern des Prozessierens ist das Erstellen eines HMM. Mit diesem wird versucht, die nicht bekannten, auf dem Wegenetz befindlichen Positionen (Kandidaten) mithilfe der bekannten, gemessenen Positionen (Beobachtungen) der Trajektorie unter Zuhilfenahme des Viterbi-Algorithmus zu ermitteln. Das Plug-in führt folgende Aufgaben sequenziell aus:

1. Einlesen der Eingabedaten und Initialisierung der internen Objektstruktur
2. Berechnung der Kandidaten
3. Berechnung der Beobachtungswahrscheinlichkeiten
4. Berechnung der Übergangswahrscheinlichkeiten
5. Anwendung des Viterbi-Algorithmus und Ermittlung der Kandidaten, die die gegebene Trajektorie am besten abbilden
6. Berechnung der abgeglichenen Strecke mit den ermittelten Kandidaten

### 2.2 Berechnung der Kandidaten

Für jede Position der Trajektorie muss mindestens ein Punkt auf dem Wegenetz ermittelt werden. Dabei wird für jede Beobachtung die kürzeste Distanz zu jedem Datensatz des Wegenetzes (Geometriotyp LINESTRING) berechnet. Ist eine berechnete Distanz nicht größer als ein vom Nutzer vorgegebener Wert (die maximale Suchdistanz), wird der zur Beobachtung nächstgelegene Punkt auf dieser Linie ermittelt und als Kandidat für diese Beobachtung definiert. Die Kandidaten werden in einem 2D-Array gespeichert, wobei die Anzahl der Spalten gleich ist mit der Anzahl an Beobachtungen. In jeder Spalte finden sich die zu einer Beobachtung zugehörigen Kandidaten (siehe Abb. 1). Lässt sich zu einer Beobachtung kein Kandidat berechnen, bricht das Plug-in die gesamte Prozedur ab und teilt dies dem Nutzer mit.



**Abb. 1:** Schematische Darstellung der internen Datenhaltung der Kandidaten analog zum Aufbau eines HMM

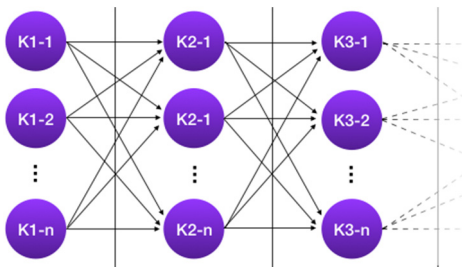
### 2.3 Berechnung der Beobachtungswahrscheinlichkeiten

Jedem Kandidaten ist ein Wahrscheinlichkeitswert (Beobachtungswahrscheinlichkeit) zuzuordnen. Dieser soll eine Aussage ermöglichen, wie wahrscheinlich dieser Kandidat die Beobachtung auf dem Wegenetz darstellt. Für die Berechnung der Beobachtungswahrscheinlichkeiten wird angenommen, dass von einer Position auf dem Wegenetz mit einer bestimmten Verteilung Positionen per GNSS ermittelt werden. Zur Vereinfachung wird postuliert, dass diese Messungen normalverteilt sind, sodass die Beobachtungswahrscheinlichkeit mit folgender Formel berechnet wird:

$$P = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (1)$$

( $x$  = euklidische Distanz zwischen Beobachtung und Kandidat,  $\mu$  = Erwartungswert bzgl. der Distanz zwischen Beobachtung und Kandidat,  $\sigma$  = Standardabweichung). Die Werte für die Standardabweichung und den Erwartungswert werden vom Nutzer vorgegeben. Die Beobachtungswahrscheinlichkeiten der Kandidaten der ersten Beobachtung entsprechenden Startwahrscheinlichkeiten bzw. der Anfangsverteilung in einem HMM, bei allen anderen Kandidaten den Emissionswahrscheinlichkeiten bzw. der Beobachtungsmatrix.

### 2.4 Berechnung der Übergangswahrscheinlichkeiten



**Abb. 2:** Übergänge zwischen allen Kandidaten benachbarter Beobachtungen

Das Plug-in bildet Übergänge zwischen allen Kandidaten benachbarter Beobachtungen (siehe Abb. 2). Um die einzelnen Übergänge vergleichen zu können, wird für jeden Übergang ein Wahrscheinlichkeitswert berechnet, wobei diese Übergangswahrscheinlichkeiten das Produkt zweier Werte darstellen.

Der erste Wert resultiert aus dem Vergleich der Distanz zwischen zwei Kandidaten und der Distanz zwischen den korrespondierenden Beobachtungen. Zur Berechnung der Distanz zwischen den Kandidaten wird die kürzeste Verbindung zwischen diesen beiden Punkten auf dem Wegenetz mittels des Dijkstra-Algorithmus, wie er über die Klasse `QgsGraphAnalyzer` der PyQGIS-API genutzt werden kann (QGIS.ORG, n. d.), ermittelt. Die Entfernung zwischen den zugehörigen Beobachtungen entspricht der euklidischen Distanz. Aus beiden Werten wird die Differenz gebildet und in folgender Formel eingesetzt (Newson & Krumm, 2009, p. 339):

$$P = \frac{1}{\beta} \cdot e^{-\frac{d}{\beta}} \quad (2)$$

( $d$  = Differenz zwischen den beiden Distanzen,  $\beta$  = Gewichtungsfaktor). Der Gewichtungsfaktor  $\beta$  ist vom Nutzer anzugeben und wird in Kapitel 4 näher beleuchtet.

Der zweite Bestandteil der Übergangswahrscheinlichkeit ergibt sich aus der Betrachtung der Richtungen zwischen den Kandidaten und den Beobachtungen. Für die Beobachtungen wird der Anstieg der Gerade, welche durch die beiden Beobachtungen aufgespannt wird, berechnet und als Winkel mit einem Wert zwischen  $0^\circ$  und  $180^\circ$  gespeichert. Für die Kandidaten wird wieder die kürzeste Strecke zwischen beiden Punkten auf dem Wegenetz genutzt. Für jeden Streckenabschnitt (bestehend aus zwei Stützpunkten) wird der Richtungswinkel berechnet und in ein Verhältnis zum Richtungswinkel der Beobachtungen gesetzt, sodass sich ein normierter Wert zwischen 0 und 1 ergibt (Budig, 2012, p. 17). Die erhaltenen Werte werden multipliziert und ergeben die Richtungswahrscheinlichkeit der gesamten Strecke zwischen den Kandidaten:

$$P = \prod_{i=1}^{\text{Anzahl Streckenabschnitte}} \frac{180 - |RB - RSA_i|}{180} \quad (3)$$

( $RB$  = Richtung zwischen zwei Beobachtungen,  $RSA$  = Richtung eines Streckenabschnitts,  $i$  = Index des Streckenabschnitts). Die Übergangswahrscheinlichkeit ergibt sich aus dem Produkt von (2) und (3). Die Gesamtheit aller Übergänge und deren Wahrscheinlichkeiten entspricht der Übergangsmatrix eines HMM.

## 2.5 Ermittlung des wahrscheinlichsten Pfades auf dem Wegenetz

Nachdem die Kandidaten ermittelt und alle Wahrscheinlichkeitsmaße berechnet wurden, liegt ein komplettes HMM vor. Durch die Anwendung des Viterbi-Algorithmus auf das HMM wird die Abfolge an Kandidaten ermittelt, bei der die Gesamtwahrscheinlichkeit maximiert wird. Die Gesamtwahrscheinlichkeit stellt das Produkt aus allen Beobachtungs- und Übergangswahrscheinlichkeiten entlang eines Kandidatenpfades dar. Anschließend wird immer zwischen zwei aufeinanderfolgenden Kandidaten die kürzeste Strecke ermittelt (vgl. Kapitel 2.4). Durch Zusammenfügen der erhaltenen Strecken ergibt sich der Weg auf dem Wegenetz, welcher die Trajektorie am wahrscheinlichsten darstellt.

## 3 Testdaten

### 3.1 Trajektorie

Zum Testen wurden mehrere manuell digitalisierte Trajektorien inklusive Ausreißern, wie sie durch Messungenauigkeiten entstehen, genutzt. Bei einer zeitgesteuerten Aufnahme der Trajektorie sollten Standzeiten im Vorfeld entfernt werden. Dies verringert die Berechnungszeit und verhindert, dass durch ein Zurückspringen der Trajektorie ein falsches Ergebnis berechnet wird. Bei einer weggesteuerten Datenaufnahme ist darauf zu achten, dass der zurückzulegende Weg zur Auslösung einer Messung nicht zu groß ist, da dies sonst vor allem im Kreuzungs- und Kurvenbereich zu ungenauen Resultaten führt.

### 3.2 Wegenetz

Als Wegenetz wurde das Straßennetz der Stadt Radebeul aus OpenStreetMap heruntergeladen (Gemeinde-, Kreis und Staatsstraßen) und als Knoten-Kanten-Modell mittels PostGIS (Funktion `CreateTopology(...)`) aufbereitet. Die Richtung der Linien ist irrelevant. Auch werden keine Einschränkungen der Fahrtrichtung beachtet. Es empfiehlt sich, dass Strecken ohne Abzweige und Kreuzungen nicht weiter unterteilt werden, da sonst für jedes Segment Kandidaten ermittelt werden, die wiederum zu weiteren Übergängen führen und die Berechnungszeit erhöhen.

## 4 Bedienung des Plug-ins

### 4.1 Installation

Das Plug-in ist im offiziellen QGIS-Plug-in-Repository enthalten und kann direkt in QGIS heruntergeladen und installiert werden. Hierfür muss über den Menüpunkt Erweiterungen bzw. Plug-ins der Plug-in-Manager geöffnet und nach „Offline-MapMatching“ gesucht werden. Es ist in allen QGIS-Versionen ab 3.0 herunterlad- und installierbar und benötigt keine Installation weiterer Python-Module. Das Plug-in verfügt über eine grafische Nutzeroberfläche und kann zusätzlich in der Toolbox und im Processing-Framework per Python aufgerufen werden.

### 4.2 Allgemeine Parameter

Als Eingabedaten sind zwingend ein Linien- und ein Punkt-Layer notwendig, wobei das Plug-in keine Multi-Geometrien unterstützt. Zusätzlich muss für den Punkt-Layer ein Feld angegeben werden, welches als Identifikator für die Datensätze fungiert. Informationen aus diesem Feld werden im Endergebnis gespeichert, um eine Zuordnung der abgeglichenen mit der ursprünglichen Trajektorie herzustellen zu können.

Es muss der EPSG-Code eines Koordinatenreferenzsystems (CRS) angegeben werden, in welchem die Berechnungen erfolgen. Dieses wird auch der abgeglichenen Trajektorie zugewiesen. Es ist zwingend erforderlich, ein metrisches CRS zu wählen, um korrekte Längen zu berechnen.

Die maximale Suchdistanz wird zum Finden von Kandidaten benötigt. Sie muss mindestens so groß sein, dass zu jeder Beobachtung wenigstens ein Kandidat gefunden werden kann. Ein zu hoher Wert führt dazu, dass sehr viele Kandidaten gefunden und Übergänge gebildet werden, was sich negativ auf die Berechnungszeit auswirkt.

### 4.3 Standardabweichung und Erwartungswert

Die Einstellungen für die Standardabweichung und den Erwartungswert beziehen sich auf die Messgenauigkeit der Trajektorie. Es empfiehlt sich, die Werte gemäß der Spezifikation des Messgerätes zu wählen. Liegen diese nicht vor, müssen die Parameter manuell gewählt werden. Ein Erwartungswert ungleich 0 ist nur dann sinnvoll, wenn Trajektorie oder Wegenetz einen gleichmäßigen Versatz aufweisen. Kandidaten mit einer Entfernung zur Beobachtung gleich dem Erwartungswert weisen eine höhere Beobachtungswahrscheinlichkeit auf als bei einer vom Erwartungswert abweichenden Entfernung (bei gleicher Standardabweichung).

Die Standardabweichung stellt die Streuung der Messergebnisse an einer Position dar. Ein hoher Wert bedeutet weniger Vertrauen in die Messungen mit dem Ergebnis, dass sich die Beobachtungswahrscheinlichkeiten bei unterschiedlich entfernt liegenden Kandidaten um dieselbe Beobachtung weniger stark unterscheiden (bei gleichem Erwartungswert). Ein zu hoher Wert verringert den Einfluss der Beobachtungs- auf die Gesamtwahrscheinlichkeit und überbetont dadurch den Einfluss der Übergangswahrscheinlichkeit. Ein zu geringer Wert führt dazu, dass ab einer gewissen Entfernung zwischen Kandidat und Beobachtung die Beobachtungswahrscheinlichkeit nahe 0 ist und somit Kandidaten ab dieser Distanz nahezu ausgeschlossen werden. Die Standardabweichung kann auch über eine Näherungsformel berechnet werden (Newson & Krumm, 2009, p. 341).

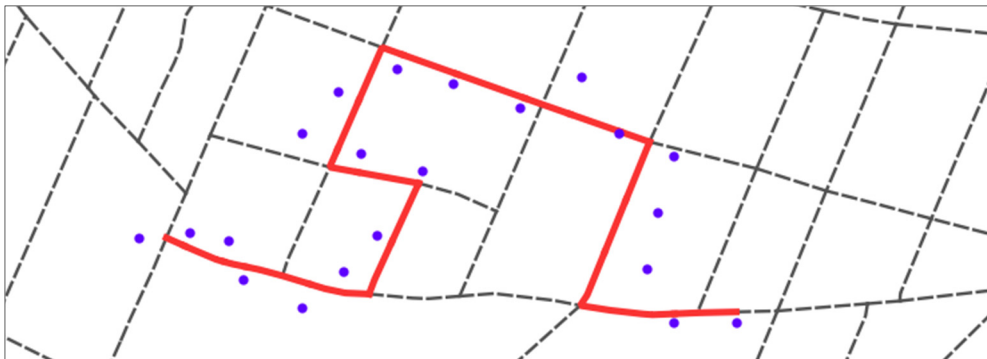
### 4.4 Gewichtungsfaktor

Der Gewichtungsfaktor ist abhängig von der Qualität der Trajektorie. Der Wert kann durch eine Näherungsformel ermittelt werden (Newson & Krumm, 2009, p. 341), sollte aber immer in Bezug zur Standardabweichung gesehen werden. Da der Gewichtungsfaktor in einer exponentialverteilten Funktion eingesetzt wird, hat dieser Wert ähnliche Auswirkungen auf die Übergangswahrscheinlichkeit wie die Standardabweichung auf die Beobachtungswahrscheinlichkeit. Ein hoher Wert erlaubt mehr Toleranz gegenüber größeren Differenzen zwischen der euklidischen Distanz zweier Beobachtungen und der Distanz zwischen den zugehörigen Kandidaten auf dem Wegenetz. Ein zu hoher Wert lässt den gesamten Algorithmus anfälliger für Ausreißer der Trajektorie werden, da dann die Übergangswahrscheinlichkeit innerhalb der Gesamtwahrscheinlichkeit unterrepräsentiert wird. Im Gegensatz dazu führt ein zu kleiner Wert dazu, dass das Plug-in möglicherweise Kandidaten ermittelt, deren Distanz auf dem Wegenetz sehr nah an der euklidischen Distanz der Beobachtungen ist, jedoch die Charakteristik der Trajektorie nicht korrekt widerspiegelt.

## 5 Test

### 5.1 Resultate in Abhängigkeit verschiedener Parametereinstellungen

Das Ergebnis des Map Matchings mit dem Plug-in ist abhängig von den Einstellungen der Parameter Standardabweichung, Erwartungswert und Gewichtungsfaktor. Der Erwartungswert wurde für Tests stets mit 0 angegeben. Da manuell digitalisierte Trajektorien für die Tests genutzt wurden, konnte nicht auf die Standardabweichung eines Messgerätes zurückgegriffen werden. Ein erfolgreiches Map Matching wie in Abbildung 3 konnte wiederholt mit unterschiedlichen Datensätzen und Parametereinstellungen erreicht werden, wobei sich die genutzten Parameter je nach Trajektorie sehr unterscheiden können. Auch können unterschiedliche Werte für die Standardabweichung und den Gewichtungsfaktor bei den gleichen Eingangsdaten zu erfolgreichen Ergebnissen führen. Daraus ergibt sich, dass die Kombination aus Standardabweichung und Gewichtungsfaktor von hoher Bedeutung für ein erfolgreiches Map Matching ist. Die Nutzung der von Newson & Krumm (2009) beschriebenen Näherungsformeln hat zu sehr guten Ergebnissen in Tests geführt und scheinen ein vielversprechender Ansatz zur Ermittlung der benötigten Parameter zu sein.



**Abb. 3:** Screenshot aus QGIS nach Durchlauf des Plug-ins (gestrichelte Linie = Straße, Punkte = Trajektorie, durchgezogene Linie = abgeglichene Trajektorie)

### 5.2 Resultate in Abhängigkeit der Daten

Die Genauigkeit der genutzten Trajektorie hat ebenfalls einen großen Einfluss. Eklatante Ausreißer sollten im Vorfeld entfernt oder korrigiert werden, damit man mit einer geringeren maximalen Suchdistanz eine deutliche Verbesserung der Berechnungszeit erlangt. Auch müssen die Ausreißer dann nicht mehr bei der Wahl der restlichen Parameter beachtet werden, was zu tendenziell besseren Ergebnissen führt. Der Abstand zwischen den einzelnen Positionen der Trajektorie sollte nicht zu groß sein. In Tests hat sich ein Abstand von 100 Metern als noch unproblematisch herausgestellt. Noch größere Abstände können im Bereich vieler Kurven und Kreuzungen dazu führen, dass die tatsächlich zurückgelegte Strecke nicht mehr genau erkennbar ist und damit das Map Matching keine korrekten Ergebnisse liefert.

## 6 Ausblick

Das Plug-in Offline-MapMatching erweitert QGIS um die Funktion, eine Trajektorie mit einem Wegenetz abzugleichen. Dabei werden die Ansätze aus mehreren Arbeiten kombiniert, um ein robustes mathematisches Modell zu erhalten. Die Qualität der Ergebnisse ist stark abhängig von den Einstellungen der Parameter. Diese müssen vom Nutzer ermittelt werden, beispielsweise über Näherungsformeln. Für eine künftige Version wird angestrebt, diese Näherungsformeln direkt in das Plug-in zu integrieren, sodass eine Nutzereingabe nicht zwingend erforderlich ist.

## Literatur

- Andrienko, N., Andrienko, G., Pelekis, N., & Spaccapietra, S. (2008). Basic concepts of movement data. In: *Mobility, data mining and privacy* (pp. 15–38). Berlin/Heidelberg: Springer.
- Budig, B. (2012). *An algorithm for map matching on incomplete road databases*. Retrieved Jul 31, 2018 from <http://www1.pub.informatik.uni-wuerzburg.de/pub/theses/2012-budig-bachelor.pdf>.
- Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W., & Huang, Y. (2009, November). Map-matching for low-sampling-rate GPS trajectories. In: *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems* (pp. 352–361). New York, NY: ACM. Retrieved Dec 20, 2018 from <https://users.wpi.edu/~yli15/courses/DS504Fall17/includes/Mapmatching09.pdf>.
- Mattheis, S. (2015). *Barefoot release – An Open Source Java library for map matching with OpenStreetMap*. Retrieved Dec 20, 2018 from <http://www.bmw-carit.com/blog/barefoot-release-an-open-source-java-library-for-map-matching-with-openstreetmap/>.
- Newson, P., & Krumm, J. (2009). Hidden Markov map matching through noise and sparseness. In: *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems* (pp. 336–343). New York, NY: ACM. Retrieved Oct 10, 2018 from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.187.5145&rep=rep1&type=pdf>.
- QGIS.ORG (n. d.). Class: QgsGraphAnalyzer. Retrieved Dec 20, 2018 from <https://qgis.org/pyqgis/master/analysis/QgsGraphAnalyzer.html>.
- Raymond, R., Morimura, T., Osogami, T., & Hirose, N. (2012). Map Matching with Hidden Markov Model on Sampled Road Network. *21st International Conference on Pattern Recognition (ICPR 2012)*. Retrieved Jul 31, 2018 from <https://www.computer.org/csdl/proceedings/icpr/2012/2216/00/06460610.pdf>.