# **Stand der Positionstracking-App**

## Funktionsübersicht:

Ein Bild, das Karte enthält.

Automatisch generierte Beschreibung Ein Bild, das Karte enthält.

Automatisch generierte Beschreibung

## Aufnahme:

**Aufnahmefrequenz**: aktuell 1 Hz (auch wenn ggf. noch keine neue Position vorliegt).

Ein Bild, das Karte enthält.

Automatisch generierte Beschreibung Ein Bild, das Karte enthält.

Automatisch generierte Beschreibung

## Datenspeicherung & MQTT:

1. Speicherung der Aufnahmen auf dem Gerät in Form von .txt-Dateien im csv-Format.
2. Senden der Daten an einen MQTT-Broker im JSON-Format. Von diesem ggf. Abfrage mit Hilfe eines Python-Skriptes (bereits umgesetzt).

 Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

# **Erste Erkenntnisse zu Forschungsfragen**

## **Formulierte Forschungsfragen:**

1. **Wie können reale Positionsdaten, die ggf. Ungenauigkeiten und Rauschen aufweisen, so vorverarbeitet werden, dass sie sich gut für den Einsatz maschineller Lernverfahren eignen?**
2. **Welche Klassifikationsverfahren sind für die Klassifizierung von Verkehrsteilnehmern auf Basis realer Positionsdaten geeignet?**

## **Zu A) / Datenvorverarbeitung:**

**Aufnahme mit der App:**

Erste Versuche mit eigenem Mobilgerät zeigen, dass Positionsdaten (GPS + WLAN&Bluetooth-Optimierung) nicht zwangsläufig stark verrauscht sind. Bisherige Versuche zeigen durchweg hohe Präzision.

Beispiel 1: Aufgenommener Fußweg von ca. 5min Länge

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Beispiel 2: Mit dem Auto gefahrene Strecke von ca. 5min Länge

Ein Bild, das Diagramm, Karte enthält.

Automatisch generierte Beschreibung

**Map Matching:**

Erste Versuche mit „Valhalla“, einer Open-Source Routing und Map-Matching Engine, die als Instanz lokal in einem Docker-Container installiert wurde. (Basis: [A Practical Guide to an Open-Source Map-Matching Approach for Big GPS Data | SpringerLink](https://link.springer.com/article/10.1007/s42979-022-01340-5)) Valhalla ist sehr mächtig und erlaubt es eine Vielzahl an Zusatzinfromationen zu einzelnen Straßen und Straßensegmenten allein auf Basis von gematchten Positionsdaten abzufragen.

Beispiel: Fahrt mit Motorrad durch Kreisel (grün = map-matched)

Ein Bild, das Karte enthält.

Automatisch generierte Beschreibung

Ergebnisse dieser ersten Versuche sind zum Teil im Großen und Ganzen vielversprechend, allerdings wurden auch bereits einige Schwächen offenbart.

Negativbeispiel 1: Verzerrung auf Grund ungenauer Karte

Ein Bild, das Karte enthält.

Automatisch generierte Beschreibung

Negativbeispiel 2: Verzerrung durch falsches Matching

Ein Bild, das Text, Im Haus enthält.

Automatisch generierte Beschreibung

Negativbeispiel 3: Starke Verzerrung durch falschen Matching-Modus

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

## **Zu B) / Machine Learning Verfahren:**

# Bereits eingesetzte Verfahren umfassen einfachere (K-Nearest-Neighbours, Decision-Tree und SVMs) und komplexere Klassifikatoren (Deep Q-Learning- & Feed-Forward-Netz), die darauf trainiert wurden die Klasse eines Verkehrsteilnehmers auf Basis einer festen Anzahl sequenziell aufgenommener Positionsdaten zu bestimmen.

**Am Beispiel der Arbeit von Dietmar Fischer:**

Aus einer Sequenz von 100 Koordinaten, die in einem zeitlichen Abstand von 0,5s aufgenommen wurden, werden zusammenfassende Merkmale errechnet (z.B. die minimale, durchschnittliche und maximale Geschwindigkeit oder Beschleunigung), die zur Klassifikation des Verkehrsteilnehmers genutzt werden. 🡪 Basis für die Klassifikation bildet also ein statisches 50 Sekunden Aufnahmeintervall.

Nur 3 Klassen: Fußgänger, Fahrradfahrer und motorisiertes Fahrzeug (keine weitere Differenzierung zwischen den motorisierten Fahrzeugen).

**Probleme bei bisherigen Ansätzen:**

* Bisher keine hinreichend guten Ergebnisse bei der Klassifizierung verschiedener Typen motorisierter Fahrzeuge.
* Aufgenommene Sequenzen, werden durch die Zusammenfassung aller 100 Koordinaten nicht als Sequenzen betrachtet, aber ggf. liegt genau hier der Schlüssel zum Erfolg.
* Es gibt im Straßenverkehr häufig Szenarien, in welchen eine klare Unterscheidung verschiedener Fahrzeugtypen anhand eines festen Intervalls nicht möglich sein wird („Motorrad beim Beschleunigen durch vorausfahrendes Auto begrenzt“, „Auto auf kurviger Landstraße hinter Fahrradfahrer“, Ampeln, Staus…). Bisherige Ansätze kommen in solchen Szenarien an ihre Grenzen.

**Idee für besseres Verfahren:**

1. Aufnahme in Intervalle fester Größe aufteilen z.B. in Zeitspannen von 10 Sekunden.
2. Die einzelnen Intervalle durch Merkmale, ähnlich wie bei Dietmar, zusammenfassen.
3. Alle Intervalle einer Aufnahme als Sequenz betrachten.

Eine Klassifizierung soll also nicht von den Merkmalen eines einzigen Aufnahmeintervalls, sondern auch von der Klassifizierung des vorherigen Intervalls abhängen.

Hierfür eignen sich insbesondere ***Rekurrente Neuronale Netzwerke***, die ihre Ausgabe zum Zeitpunkt t als zusätzliche Eingabe im Zeitpunkt t+1 zur Verfügung stellen.

Erste Recherchen zu RNNs in diesem Kontext erweisen sich als vielversprechend, allerdings sind evtl. gegebene Nachteile aktuell noch nicht absehbar (z.B. wäre es möglich, dass mehr Trainingsdaten als bei anderen Ansätzen benötigt werden).

Eine Arbeit, welche einen solchen Ansatz verfolgt ist: [Vehicle classification from low-frequency GPS data with recurrent neural networks - ScienceDirect](https://www.sciencedirect.com/science/article/abs/pii/S0968090X18304017?via%3Dihub).

# **Datenbasis**

**Nutzbare Datensätze?**

Erste Recherchen leider ergebnislos.

**Datensammlung über Positionsapp:**

*Aufnahmefrequenz:*

Die App erlaubt aktuell das gelabelte Aufnehmen der eigenen Positionsdaten über einen beliebigen Zeitraum mit einer **Sampling-Frequenz von 1 Hz**. Dies ist unabhängig davon, ob nach einer Sekunde tatsächlich bereits eine neue Position vorliegt oder nicht (selten Dopplungen auch bei Bewegung des Geräts).

Ein Versuch mit eigenem Android-Smartphone hat gezeigt, dass eine **Erhöhung dieser Frequenz scheinbar nicht möglich** ist. Eine Verringerung führt dazu, dass der Informationsgehalt der Positionsdaten potenziell sinkt (wenn diese sehr genau sind). Bei etwas weniger genauen Positionsdaten könnte eine Verringerung der Aufnahmefrequenz jedoch auch zu einer relativen Verringerung des „Verrauschungsfehlers“ dieser Daten führen, insbesondere wenn die Position waagerecht zur Bewegungsrichtung verrauscht ist.

**Datenspeicherung**:

Jede Aufnahme wird direkt auf dem Gerät (im Downloads-Ordner) als .txt-Datei im csv-Format gespeichert. Zusätzlich werden die Daten während der Aufnahme im JSON-Format an einen MQTT-Broker gesendet, von welchen Sie (zeitgleich!) auch abgefragt werden können.

Aktuell wird eine Abfrage der Daten durch ein Python-Skript auf dem Privatrechner umgesetzt. Nach der Abfrage werden diese auch hier ins csv-Format umgewandelt (besser wäre ggf. das Speichern in einer Datenbank oder Zeitreihendatenbank).

**Datenerzeugung:**

Autos, Motorräder, Fahrräder und Fußgänger.

LKWs?

# **Anmeldung der Bachelorarbeit**

**Themenformulierung / Titel:**

***Klassifikation von Verkehrsteilnehmern auf Basis realer [Positionszeitreihen | Positionsdaten] mit Verfahren des maschinellen Lernens.***

**\* Beschränkung der Untersuchung auf RNNs**

**\* allgemeinere Betrachtung (ggf. zu umfangreich für BA)**

**Betreuer:**

**Prof. Dr. Thomas Wieland**

**Art der Veröffentlichung:**

volle Veröffentlichung