

# Kurzfassung

Ziel dieser Thesis ist es, zu ermitteln, welche Nachbearbeitungsschritte notwendig sind, um fehlerfreie und vollständige Mobilitätsdaten von Wegen, die mit Hilfe von Smartphones und GPS-Daten aufgenommen wurden, zu erreichen. Dabei werden primär Methoden untersucht, entwickelt und soweit möglich in einem Python-Programm implementiert, welche die automatisierte Plausibilisierung, Aufbereitung und Auswertung von Mobilitätsdaten erlauben. Die Daten werden dabei mittels Smartphone und der MultiMobil-Applikation erhoben.

Die MultiMobil-App wurde an der Bergischen Universität Wuppertal im Jahr 2017 entwickelt und im Projekt ÖPNV 2017/18 von Studenten zur Aufnahme von Mobilitätsdaten genutzt. Die aufgezeichneten Wege sind zu Fuß und mit Verkehrsmitteln wie Bus, Bahn und PKW zurückgelegt worden. 46 der 50 Projektteilnehmer haben einer Nutzung der erhobenen Wege in dieser Thesis zugestimmt. Der Datensatz umfasst 1123 Wege und dient in dieser Thesis als Quelle für Mobilitätsdaten und zum Testen der entwickelten Methoden zur Plausibilisierung, Aufbereitung und Auswertung der Wegedaten. Den Teilnehmern wurde zudem eine Umfrage zur Nutzung und zu deren Erfahrungen mit der MultiMobil-App gestellt, deren Auswertung im Rahmen der Thesis erfolgte und die als Grundlage für die Erarbeitung von Verbesserungsvorschlägen für die MultiMobil-App diente.

Im Grundlagenteil der Arbeit werden die technischen Möglichkeiten der GPS-gestützten Wegeerfassung sowie verfügbare Geoinformations-Anbieter vorgestellt. Anfragen an diese Anbieter werden von dem erstellten Nachbearbeitungsprogramm zur nachträglichen Generierung von vergessenen Wegen genutzt. In einem eigenen Kapitel wird die MultiMobil-App und das Projekt ÖPNV 2017/18 beschrieben.

Thematischer Schwerpunkt ist die Identifizierung der bei der Aufzeichnung mit der MultiMobil-App auftretenden Fehlerarten und die Entwicklung von Methoden zur automatisierten Korrektur dieser Fehler. In dieser Thesis wurde ein Python-Programm entwickelt, welches die meisten der vorgestellten Lösungsansätze beinhaltet und für die Aufbereitung von Wegedaten genutzt werden kann. Die Funktionen und Erweiterungsmöglichkeiten des entwickelten Nachbearbeitungsprogramms werden in einem eigenen Kapitel erläutert.

Ein wichtiger Schritt zu vollständigen Mobilitätsdaten ist die nachträgliche Ermittlung von Wegen, deren Aufzeichnung von Benutzern der MultiMobil-App vergessen werden. Dafür wurde eine Methode zum Routen dieser Wege entwickelt und in das Python-Programm implementiert. Grundlage hierfür sind die Einträge von Probanden in ein Nachtragsformular und Daten, die von den Routing-Dienstleistern Mapbox und VRR bereitgestellt werden. Ein Vorschlag, wie diese Methode auch zum Schließen von größeren Lücken in aufgezeichneten Wegen genutzt werden kann, wird diskutiert.

Im Hinblick auf das Ziel der Arbeit ist zu sagen, dass mit den im Rahmen dieser Bachelorarbeit zur Verfügung stehenden technischen Lösungen nach entsprechender Nachbearbeitung nahezu fehlerfreie und vollständige Mobilitätsdaten aus den mit der MultiMobil-App aufgenommenen Wegen gewonnen werden können.

## Abstract

The goal of this thesis is to study which post-processing steps are needed to achieve errorless and complete mobility-data which are recorded by the smartphone-application “MultiMobil”. Primarily the needed steps after the recording of such data were investigated, namely the plausibility-check, data-enhancement and data-analysis. According methods have been implemented into a Python-program.

The MultiMobil-application has been developed at the Bergische Universität Wuppertal in 2017. The application differentiates between the following means of transport: by foot, bike and Bikesharing, car and carsharing, taxi, train, local train, metro, tram and bus. During the project “ÖPNV” in the winter term of 2017/18 the application was used to record mobility-data of 50 project-participants (students) during a two-week survey period. 46 students allowed the usage of their 1123 recorded routes in this thesis. Due to these routes, the testing of the developed Python-program was greatly simplified. To find issues of the MultiMobil-app and to get recommendations for possible enhancements of the app the participants of the “Projekt ÖPNV” have been given a query, which evaluates the way of usage of the MultiMobil-app as well as encountered problems with it.

The theoretical part of this thesis is divided into three chapters. The first discusses the technical requirements for GPS-based tracking and compares different geo-information providers, three of these are later used in the data-enhancement process. The second part describes the MultiMobil-app and the project “ÖPNV”. In the third part, the programming language Python and the used Python-modules are presented.

The main chapters of this thesis cover the identification and resolution of errors which occur during the recording of mobility-data with the MultiMobil-app. To achieve this, a computer program in the programming language Python, which detects and fixes those errors, was developed. The main correction function detects deviating points in a route and removes them automatically.

One major issue of the MultiMobil-app are missing routes, due to the fact, that participants need to start the recording of each route by themselves. Therefore, a big challenge was to close those exiting gaps in the route-set. As a solution, those missing routes are routed automatically by the Python-program based on supplementary forms filled by the participants and routing data provided by the routing-providers “Mapbox” and “Verkehrsverbund Rhein Ruhr” (VRR). The possibility to use this solution to close large gaps in existing routes is also discussed.

The realized program detects most of the found error-types. Nevertheless, further improvements of the program are discussed in a distinct chapter.

The drawn conclusion of this thesis is that by according post-processing with the developed Python-program significant improvements in data-quality as well as a considerably reduction in workload of its users can be achieved.