Multimodal passive tracking of passengers to analyse public transport use

Alessio Daniele Marra

Henrik Becker

Kay W. Axhausen

Francesco Corman

ETH Zürich

May 2018

STRC | 18th Swiss Transport Research Conference
Monte Verità / Ascona, May 16 – 18, 2018

ETH Zürich

Multimodal passive tracking of passengers to analyse public transport use

Alessio Daniele Marra, Henrik Becker, Kay
W. Axhausen, Francesco Corman
Institute for Transport Planning and Systems
ETH Zürich
Stefano-Franscini-Platz 5 8093 Zurich,
Switzerland
phone: +41 44 633 24 15
{alessio.marra,henrik.becker,axhausen,francesco.corman}@ivt.baug.ethz.ch

May 2018

Abstract

We tackle the problem of understanding multimodal passenger flows to better address mobility. In particular, we use large scale, long-term GPS passive tracking of travellers, to improve public transport operations in urban areas. We combine this with realised operation data from a transit operator. We develop an unsupervised tool able to detect passengers' travel behaviour and the exact public transport means they took, without any assumption on the frequency of GPS data. Furthermore, we show the significant advantage of using past user data to improve the detection algorithm, in particular to identify the points where users transfer. We test this approach in Zurich and refer to the multimodal mobility supply available there. Ultimate goal of this ongoing work is to understand from realised mobility how passengers react to disturbances, especially for working days and peak hours, and to the mitigation action undertaken by operators in case of disturbances, in order to design good and effective mitigation actions.

Keywords

Tracking, travel survey, public transport operations, GPS, activity identification, transportation mode detection, vehicle imputation

1 Introduction

Urban areas are related to large amounts of inhabitants and economic value, and this trend is increasing. Mobility is a necessary service, for which people have stringent economic and quality constraints, and increasing service level targets. To achieve both, the entire mobility supply has to be considered, where public transport and collective mobility might have a large role. To understand some of those trends, travel diaries have been the primary source of information on the dynamics of travel behavior capturing activity chains, trip patterns, mode choice and time use. Due to their high response burden, traditional pencil-and-paper or telephone-interview surveys (which both are still widely employed) typically ask respondents to report one randomly chosen day only. Moreover, a lot of efforts and time are required to conduct them and they represent only a partial view of the whole dynamic behaviour. However, it has already been shown that intra-person variability constitutes a substantial share of overall variation in travel behavior (Pas and Sundar (1995)), which limits the political and operational value of one-day data (Jones and Clarke (1988), Susilo and Axhausen (2007), Schlich and Axhausen (2005)).
With this study, we aim to understand the behaviour of people by long-duration passive tracking, without requiring any efforts from them and without affecting their daily life. In this way, the use of locational data provided by smartphones can reduce drastically the efforts to conduct travel surveys, widening the scale of the observations and leading to a better representation of the whole phenomenon. One of the drawbacks of smartphone-based tracking systems is that energy-intensive GPS services may affect the battery cycle of the smartphone. In fact, a high frequency of position requests can drastically shorten the battery life, making it impossible to track users for a longer time. To overcome these problems, we designed a low-battery consuming smartphone application that passively collects location information with a low sampling frequency, invisible to the user. Then, we designed different algorithms to understand the user's travel behaviour and to deal with low-quality locational data. In particular, we focused on activity and trip identification, mode detection and vehicle imputation. The proposed mode detection algorithm does not require any additional information from the users, it is completely unsupervised, it does not rely on a statistical inference model and it is also able to identify the exact public transport means a user took.
The key contributions are:

- Multi-modal transport operations are considered in a fully integrated way
- Large scale, long duration passive tracking data of travellers are collected
- User's travel behaviour is derived from low-frequency GPS data, collected by a smartphone application invisible to the user
- Mode detection and vehicle imputation are estimated by cross analysing tracking data with realized transit operations

- User's past travel information are used to improve the mode detection

The data collection of our study is based on a period of four weeks, during which the smartphone applications passively collected users' location information approximately every 30 seconds. During this period, no respondent interaction was required apart from a short questionnaire and installation of the app. Moreover, we collected the realised public transport operations data of the city of Zürich during this period. Then, activity detection, trip segmentation, mode choice and vehicle imputation are performed in a completely unsupervised manner. The rest of this paper is organized as follows: in chapter 2, the main related works are described, even if a short literature review is presented at the begin of each chapter, as each step of the GPS data processing can be seen as unrelated to the others. In chapter 3, the smartphone application used, the survey process and the whole dataset are described in detail. In chapter 4, the data cleaning procedure, performed before analysing the GPS data, is discussed. Chapter 5 presents the trip and activity identification algorithm. Chapter 6 describes the trip segmentation algorithm, that divides each user's trip in walks and stages. Chapter 7 shows the mode detection algorithm and the final results. Then, in chapter 8 there are the conclusions.

2 State of the art

GPS loggers opened new doors to collect longer-term travel diaries at a substantially lower response burden (Stopher *et al.* (2008)). Data collected with GPS loggers mounted in private cars also showed that traditional survey methods were prone to substantial under-reporting of trips (Bricka and Bhat (2006)) limiting the validity of their results. First studies using personal GPS loggers for travel diary collection were very promising, although a substantial administrative effort was required for distribution of the devices and a prompted recall to obtain additional information necessary to interpret the GPS records (Bohte and Maat (2009); Oliveira *et al.* (2011); Schuessler and Axhausen (2009); Montini *et al.* (2014)). Lately, the focus has shifted away from GPS loggers towards smartphone apps to collect travel diaries (Cottrill *et al.* (2013)). Besides the benefit of an easier survey administration, respondents are also less likely to forget to take their smartphone along for a trip (compared to GPS loggers). Moreover, various systems have been developed for the transport mode detection based on GPS data collected by smartphones (two related literature reviews are Wu *et al.* (2016), Nikolic and Bierlaire (2017)). So far, most smartphone-based tracking systems use a prompted recall approach: in this setting, respondents are asked to manually add further details such as trip purpose, mode, group size, transit fare, parking fees etc. to each trip. Although some systems already employ forms of statistical learning to make suitable suggestions, a substantial amount of user interaction to annotate or validate trip information is still required. Other approaches based on passive tracking,

with particular focus on transit, have been focusing on smart card data. In those cases, mode choice, vehicle choice, and even timing cannot always be precisely established (see for example the review of Pelletier *et al.* (2011)).
So far, most of these methods have been tested on small datasets: for instance, Tsui and Shalaby (2006) collected 60 trips with the help of students; Stenneth *et al.* (2011) recorded information about three weeks for six people. The only works based on a dataset of considerable size, rely on dedicated GPS devices, making the collection of the data more complicated and expensive. Zheng *et al.* (2010) collected information of 65 people for a period of 10 months and Schuessler and Axhausen (2009) build a dataset from 4882 people (requiring multiple waves to limit the number of devices).
Among the studies concerning the automatic mode detection from GPS data, some authors (Schuessler and Axhausen (2009), Stopher *et al.* (2005), Zhu *et al.* (2016), Zheng *et al.* (2010), Zhang *et al.* (2011)) addressed transport mode detection by dividing the problem in different sub-tasks (with small variations among them):

- Data Cleaning
- Trips and Activities detection
- Trip segmentation
- Mode detection

Also this paper follows this structure. Since these tasks are often considered as separate problems, at the begin of each section a short literature review is introduced.

3 Smartphone Application and Dataset

To allow collection of long-term travel diaries with minimal respondent interference, a new smartphone app, the *ETH-IVT Travel Diary*, was developed. The app was tested in a field trial with students at ETH Zurich. Analyses in the following sections are based on the records from this field trial.

3.1 App Design

As described above, the primary purpose of the app is to collect travel diary data during multiple weeks or even longer periods of time. Hence, to keep response burden at an acceptable level, the app has to be minimally intrusive in that it must not require regular interaction with the respondent or substantially affect the device's battery life. With respect to battery life, this means

that the sampling frequency cannot be set arbitrarily high. Yet, because data is acquired during a long duration and because travel behaviour follows regular patterns (Susilo and Kitamura (2005)), lower sampling rates can be afforded.
Taking into account these design considerations, a passive tracking app with minimal user interaction was developed. Respondents can download it from Google Play store[1]. The user interface only consists of a brief description of the study, a field to enter the respondent's individual ID and a button to start data collection. Once launched, data collection is performed in a background process, which remains active until the end of the study period. Even after re-start of the smartphone, data collection will resume automatically. During the study period, a notification is displayed reminding respondents of the ongoing data collection. At the end of the study period, the app shows a notification to remind the respondent to uninstall the app.
As direct access to the device's GPS is not possible, *ETH-IVT Travel Diary* requests location data from Android's internal location services. To reach an optimal tradeoff between data quality and battery consumption, a three-layer approach was used

- *high-priority* requests every 45 seconds,
- *low-priority* requests every 15 seconds,
- *zero-priority* requests every second.

It is important to note that update of location information is at discretion of Android's location services. Typically, location is determined using GPS, Wi-Fi and Bluetooth. Update frequency and sensors used usually depend on the frequency and priority of location requests of all apps and varies by operating system version and device. Hence, records are usually more accurate and precise when respondents use fitness trackers or navigation apps in parallel. The app makes further use of such data by the *zero-priority* requests, which skims the latest available location information without triggering an update.
Data was uploaded to a secure server about every six hours. However, in contrast to prompted-recall approaches employed by earlier studies, no interface is provided for respondents to (re)view their records. To allow for flexibility in the survey design, study duration, upload frequency, sampling frequencies and priorities can be configured remotely.

3.2 Data Collection

To complement the travel diary data with additional information on respondents and collect their feedback to the app, a three-step survey set-up was designed. It consists of three parts:

[1]An iOS version of the app is under development.

1. entrance survey capturing information on socio-demographic background, mobility tool ownership and attitudes;
2. four-week travel tracking using the app;
3. (optional) exit survey to collect feedback and for validation of selected records.

Students enrolled in civil engineering at ETH ($N = 1209$) were invited to the field test via e-mail in late March 2018. They were promised an incentive of CHF 20 for their participation. No reminder was sent. 102 students signed up for the study, of which 63 hold an Android smartphone and were therefore eligible for the first field trial, which 48 respondents completed. However, for 9 of those respondents, data quality did not allow further analyses. Hence, travel diaries of 39 respondents were used in this paper. They were complemented by the traces of 2 of the co-authors. At the end of the study, 35 respondents completed the exit survey and provided feedback to the app. In total, 1 032 days' worth of travel diaries were collected, which corresponds to an average of 25.2 days per respondent.
Data quality varied substantially between different users. Further analyses will be conducted to study the effect of smartphone user types and devices. Yet, user-friendliness was generally rated high among respondents with 80% stating that battery consumption was acceptable. This is an acceptable result given that apart from performing the tracking, the app requires respondents to have their GPS turned on at all times, which in itself already increases battery consumption.

4 Data Cleaning

GPS technology is widely used in different devices to capture the most accurate position for the users. Nevertheless, it is often error-prone and then it needs a pre-processing phase to correct it. In this study, position data are collected by smartphone, as described in Section 3, so they have neither a specific and rigorous sampling frequency nor a minimum value of precision, so they can vary significantly for different users and moments. Moreover, there are different possible sources of error: satellites, atmospheric disturbances, bad weather, manufacturing defects of the smartphone, urban canyons, insulating places (trains or basements) or also the user. In fact, the user is able to turn off/on the GPS, the application or the smartphone. These different sources can affect the accuracy of each GPS point and consequently the precision of the whole user's tracked path. Two major techniques are applied in literature to identify and correct GPS errors: Data filtering, used to remove systematic error, while smoothing is used to reduce random error. They are applied together or alone: Schuessler and Axhausen (2009) used a first phase of filtering and then a phase of smoothing, Zhang *et al.* (2011) used only smoothing, while Ansari Lari and Golroo (2015) only filtering. In this study, there is a first phase of data filtering and then a phase of smoothing.

4.1 Data Filtering

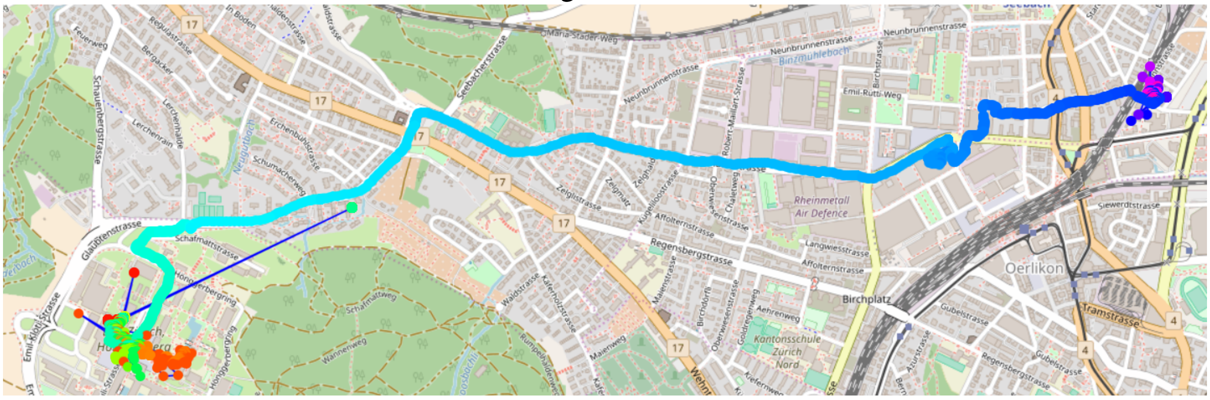
The filtering process is used to remove all the erroneous data that do not represent the real position of the user. In previous studies, different characteristics of points were used to discern between a real and a erroneous one. Ogle *et al.* (2002) demonstrated that the position dilution of precision (PDOP) and the number of satellites are useful for the filtering process, nevertheless in this study only position, time and a speed (derived from position and time) of points are available. An accuracy value in terms of meters is also available, but it has not been considered reliable for the filtering process. Among the different studies concerning GPS data processing, Schuessler and Axhausen (2009) used altitude values and sudden jumps in position to identify bad points. Ansari Lari and Golroo (2015) established that speed is a significant discriminant for GPS reliability, so they used a maximum speed value for each transport mode to filter the data. Gong *et al.* (2012) discard points if horizontal dilution of precision (HDOP) > 4 or number of satellites < 4. Instead, Stenneth *et al.* (2011) applied a manual filtering step, removing all GPS points with low accuracy or unrealistic changes in speed.
Here two main features are used to filter erroneous GPS points: speed and the angle between points. For each point the speed is calculated from the time and the position of the previous point recorded for the user. Points with speed equals to 0 are removed, because it is probable the smartphone merely returned the previous recorded position and not the real one. Furthermore, points with speed >= 150 Km/h are filtered, as the maximum speed accepted. A second feature considered, that has never been used in the literature reviewed, is the angle between points. In particular, a point that forms a very strict angle with the following and previous points and which is far away from the previous, is considered as fake GPS point that does not represent the real path of the user, but more probably it is a consequence of the loss of signal, urban canyons or other sources of error. For this reason, the following rule including the angle is used to filter these points: all points with an angle $< 15°$ and a distance from the previous point > 60 meters are removed. This rule is applied iteratively to all user's points, until any point is removed. The distance threshold was found empirically and it is used because a small angle between near points is possible and more frequent if the user is walking or just staying in the same place. Instead, in the rare case of a user moving more than 60 meters and coming back to the same place, in a time short enough to record only three GPS points, the removal of the second point and the loosing of the information of this movement will not affect the following algorithms. In fact, the user is considered to perform the same activities, walks and stages with or without this additional movement. Fig. 1 shows the application of the data cleaning process to a user's day, recorded with a high sampling frequency ($\approx$ 1 second). In particular, it is possible to see that some erroneous points in the left part of the path are removed, because of their strict angle.

4.2 Data Smoothing

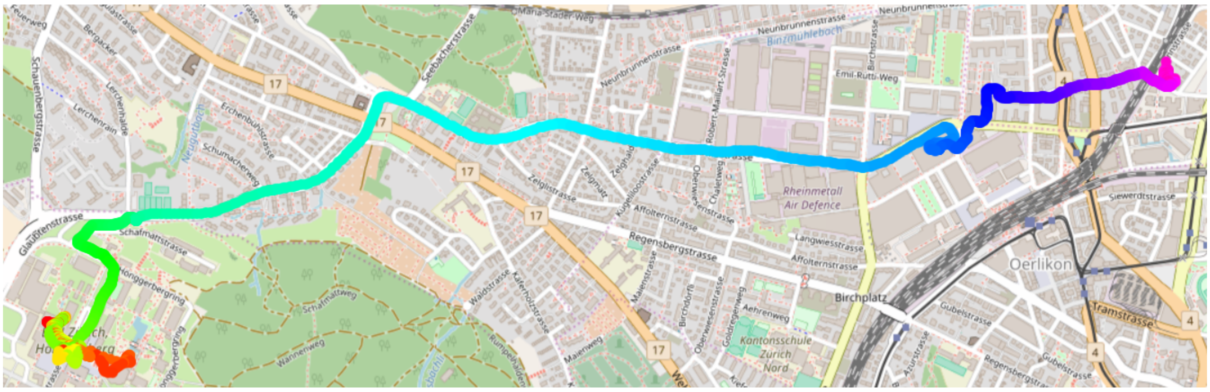
The smoothing process is used to remove the random error and the noise present in the data. Jun *et al.* (2006) compared three different smoothing techniques to minimize the impact of random error in GPS data: least squares spline approximation, kernel-based smoothing and (modified) discrete Kalman filter. Their results showed that the best method is the Kalman filter. Nitsche *et al.* (2014) also preprocessed the positional data using a Kalman filter. Instead, Schuessler and Axhausen (2009), calculated speed and acceleration from the position and the time of the GPS points, then it used a Gauss kernel smoothing approach for smoothing GPS position with three coordinates (x,y,z). Chung and Shalaby (2005) applied a moving average approach to smooth the speed associated at each point, with the aim of preventing the influence of noisy speed values during the following identification of mode change points.
Here, after the data filtering phase, we applied a Kalman Filter for the smoothing process of the two space coordinates (longitude and latitude) of the GPS data. This type of filter is able to deal with inaccurate observations and its efficacy increases with the frequency of the observations. In our study the sampling frequency of the users' GPS points is set to a low value (30 seconds) and it is variable and depends on different factors like the OS of the smartphone, the other applications running in background and the user's location. For these reasons, the impact of the smoothing process is different for different users and paths. In particular, with a sampling frequency of 1 second, wrong points can be completely corrected, instead with a frequency greater than 30 seconds, only small adjustments of the user's trajectory are applied. An example of the effect of the whole data cleaning process can be seen in Fig. 1. In this case, the smoothing process brings closer together the points in the bottom-left and top-right corner of the figure, that correspond to locations where the user is not moving for a long time.

Figure 1: Data Cleaning: Comparison of original data with cleaned data. Colour represents the time (from red to purple).

Original Data



Cleaned Data



5 Trip and Activity Identification

As the smartphone application is always active, it records the user's position continuously through the day. Therefore, each user's daily data have to be divided in trips and activities. An activity is a sequence of points near to each other, indicating that the user is in the same place for a long period. Instead, a trip is a connection between two activities, representing a movement of the user to a different place. In that way, a user's day is formed by activities alternating with trips, and these can be used to analyse the movement of the user and for the mode detection.

Different techniques have been applied in literature for this problem. So far, all of them are based on the detection of the activities and consequently the trips. One of the most common is to measure the dwell-time, the time between two consecutive GPS points, and comparing it to a certain threshold value. Nevertheless, this threshold is not fixed and it varies between 45 seconds (Pearson (2001)), 300 seconds (Wolf *et al.* (2004)) and 900 seconds (Schuessler and Axhausen (2009)). A second criteria is to identify an activity when there are very low values of speed for a certain minimum amount of time (Tsui and Shalaby (2006)). For instance, Schuessler and Axhausen (2009) consider a speed smaller than 0.1 *m/s* and a minimum duration of 120 seconds.

A third criteria is the density-of-points based method (Stopher *et al.* (2005), Schuessler and Axhausen (2009)). Since the recording of the data is continuous as long as the application is active on the smartphone, when a user is standing in one place the application registers a bundle of points near the same place. For that reason an activity can be detected where the density of points in a certain area is greater than a specific threshold. Schuessler and Axhausen (2009) computed a value of density for each GPS point by counting how many of the 30 preceding and succeeding points are within a 15 meters radius. An activity occurs when there is a sequence of points with a density higher than 15 for at least 10 points or 300 seconds.

In this paper, a density-of-point based strategy is used, which has some important differences from those mentioned above. The major one is that the algorithm does not rely on the number of GPS points (like Schuessler and Axhausen (2009)) or on their frequency. In fact, the algorithm can not rely on them, because the frequency of the acquired data is variable due to different factors as the location, the other application in background or the interaction between user and smartphone. For that reason, we define an activity when there are at least 2 successive points for at least 15 minutes in a radius of 250 meters. We choose 2 points because with 1 point it is not known if it is an activity or if no signal has been received. More formally, an iterative algorithm was developed, iterating on each point P: if there are following points in 250 meters of radius for at least 15 minutes, P is the starting point of the activity and the last point in the radius is the ending point. The radius, for each point Q following P, has the center in the average point of all the points from P to Q (excluded).

We choose a very high radius compared to Schuessler and Axhausen (2009) and Stopher *et al.* (2005) (15 and 30 meters), to better deal with the low precision of the GPS data and the position jumps that eventually can occur. Nevertheless, very short walks, starting and ending near an activity, can be considered as part of the activity, because they are still in the radius. It is possible that the first and last points of an activity are in reality points of the previous or following trip, so two rules are added to the algorithm to better assign these extreme points. After that an activity is found, the middle point is computed as the average of the coordinates of its points.

- If the distance from the starting point to the middle point is greater than two times the average distance of each point to the middle point, an activity is not identified and the algorithm tries from the next point.
- If the distance from the end point to the middle point is greater than two times the average distance of each point to the middle point, check the second-last as the ending point.

With these additions, the break points between activities and trips can better be detected and the mode detection algorithm will benefits from this. Fig. 2 shows an example identification of an activity. In particular, the first and last points, relating the user's arrival and departure, are not included in the activity, even if they are in the radius and they are not the farthest from the center.

Figure 2: Activity identification: The red points are in the activity, the green points are the last points of the previous trip, the blue points are the first points of the following trip.

Furthermore, in case of multiple erroneous recording of GPS position, caused often by detecting the position near cell sites or antennas, an additional rule is added to avoid the detection of false positive trips: a trip with origin and destination closer than 250 meters and with a duration less than 5 minutes, is merged between the previous and following activity to one single activity.

6 Trip Segmentation

The trip segmentation process consists of partitioning of the users' trips in walks and stages. A walk occurs when the user is walking or he is just waiting for a means of transport in a single place, while a stage is a movement of the user done with the help of a transport means, that can be a car, a bus, a train, a bicycle or other vehicles. This step is the most challenging part of the mode detection process, in fact, there is not a common solution in literature to address this problem and different studies used different sensors or they rely on a high sampling frequency for the GPS data.

The most relevant feature, on which all the trip segmentation algorithms found in literature are based, is speed. For instance, Biljecki *et al.* (2013) considered the stops of the user as potential transition-points between two different modes. A stop is detected when consecutive points in an interval of 12 seconds do not have a speed higher than 2 km/h. Gong *et al.* (2012) applied several rules to identify a walk segment, and most of them rely on the speed of the points. Three of them are: the first point has a speed $\geq$ 1.6 km/h and $<$ 10 km/h; minimum duration of 60 seconds; 85th percentile of speed of all points ≤ 10 km/h. Other works are also based on acceleration to detect walks or stops (Shin *et al.* (2015), Zheng *et al.* (2010)). Between two different transport modes the user usually walks or stops for a certain period, so Zheng *et al.* (2010) considered a threshold of speed and of acceleration to divide the points in walks and non-walks, then it merges segments of points of the same type according to rules depending on the length of the segments. A paper using a similar procedure is Zhu *et al.* (2016), that applies a label specification step to mark the points as walk or non-walk according to two threshold values of speed and acceleration. Then the label of each point is adjusted according to the near points: if at least M (value dependent by the number of points) of the previous and posterior points have a different label, the label of the point is changed. Zhang *et al.* (2011) identifies different stages looking for stops of the user. It considered the heading change as a main parameter for the identification of stops, because stops are usually accompanied with large magnitude values in heading changes. Its algorithm relies on three different rules about changes in magnitude of heading, position and velocity, but it is based also on a frequency of 1 second for acquiring position, that is relatively high. Furthermore, Liao *et al.* (2006) used also GIS information for trip segmentation, in particular the proximity to transition locations such as a bus stop. However, this information can not be used in this case, because Zürich has a high density of bus stops and the GPS error can be very high.

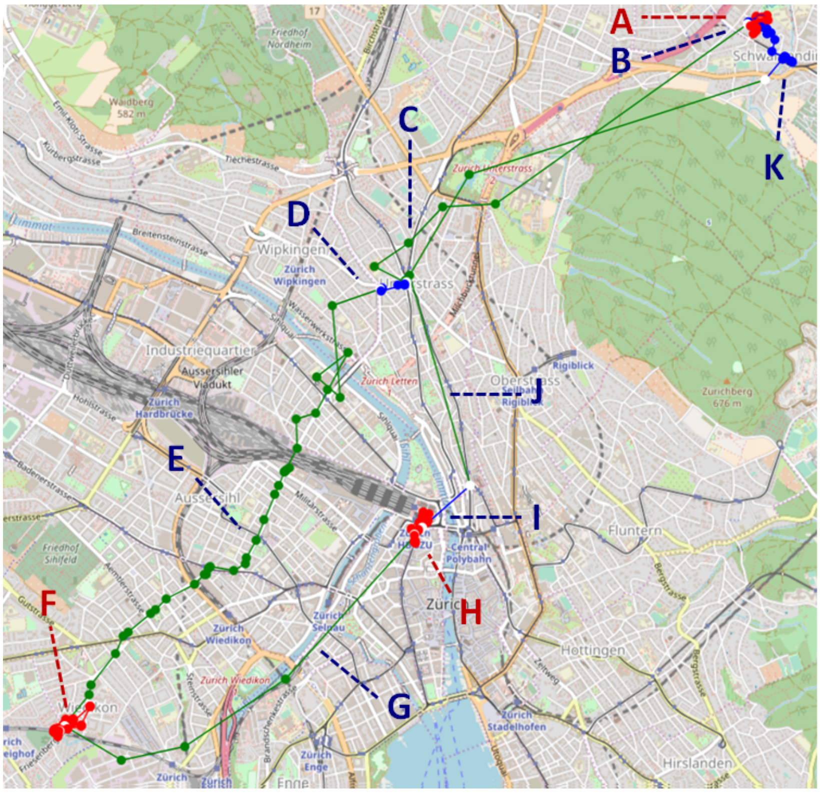
The GPS data recorded for this study have a low sampling frequency that is not fixed and it can vary considerably with different paths or users, so the algorithm developed can not rely on the frequency or on the number of points. Furthermore, the heading change and a derived acceleration are not usable for this purpose with this sampling frequency. A specific segmentation algorithm able to deal with irregular sampling frequency and based only on the GPS position and a derived speed had to be designed. Before computing the trip segmentation for each trip, if in a trip there is an absence of signal for more than 10 minutes, it is divided in two different trips. This time is considered as a threshold to avoid error during the segmentation. Our trip segmentation algorithm aims to divide a trip to a sequence of walks and stages (transfers are considered as walks). It is based firstly on a label specification step, similar to Zhu *et al.* (2016), but considering the adjacent points according to time and not to the number of points. Then, the consecutive points of the same type are grouped in sequences, and these latter are merged according to rules depending on duration, distance and speed. The segmentation algorithm is described in Fig. 3. The parameters *minSpeed*, *maxTime* and *scale* are used for the label specification and they are set respectively to 8.7 *Km/h*, 60 *seconds* and 0.8. The parameters *minDuration*, *minDistance*, *stageMinDuration* and *walkMinDuration* are used to merge small segments in walks and stages and they are set respectively to 20 *seconds*, 350 *meters*, 50 *seconds* and 70 *seconds*. These values have been tuned empirically, in particular: *minSpeed* has been set greater than Zhu *et al.* (2016) (6.48 Km/h), to reduce the number of wrongly detected stages, because on the contrary wrong walks can be corrected by the mode detection

Figure 3: Trip segmentation algorithm: Label specification (lines 2:19), merging rules (lines 22:25, 26:29, 30:33)

```
1: procedure SEGMENTATION(trip, minSpeed, maxTime, scale, minDuration,
   minDistance, stageMinDuration, walkMinDuration)
2:     for each point ∈ trip do
3:         if point.speed < minSpeed then
4:             point.type ← walk
5:         else
6:             point.type ← stage
7:         end if
8:     end for
9:     repeat
10:        for each point ∈ trip do
11:            adjacentPoints ← previous and next points in maxTime
12:            M ← size(adjacentPoints) ∗ scale
13:            if more than M points in adiacentPoints are walk then
14:                point.type ← walk
15:            end if
16:            if more than M points in adiacentPoints are stage then
17:                point.type ← stage
18:            end if
19:        end for
20:    until no changes
21:    segments ← group all sequence of points of the same type
22:    repeat
23:        merge each segment with duration D < minDuration between
24:        previous and next segment if their duration > D
25:    until no changes
26:    repeat
27:        merge a stage between two walks if its distance < minDistance
28:        or its duration < stageMinDuration
29:    until no changes
30:    repeat
31:        merge a walk between two stages if its average speed > minSpeed
32:        or its duration < walkMinDuration
33:    until no changes
34:    return segments
35: end procedure
```

algorithm; moreover, is not common that a user's stage is shorter than 350 meters (*minDistance*), lasts less than 50 seconds (*stageMinDuration*) or that a user's walk is shorter than 70 *seconds* (*walkMinDuration*). The whole algorithm principally relies on the speed calculated at each point, while other information like the acceleration and the heading change can not be used.

Figure 4: Trip segmentation of a user's day (from A to K): activities (red), stages (green), walks (blue). The table gives the average time between two points (sampling frequency) for the segments and the activities.



	Average Time (s)
A	76
B	85
C	56
D	29
E	21
F	25
G	124
H	53
I	312
J	264
K	53

For this reason, there are few cases in which the trip segmentation will fail: a fast run can be detected as a stage; a rapid change of bus, with the second one departing quickly after the arrival of the first one, can be detected as a single stage; a vehicle stuck in traffic for a long time can be detected as a walk. To overcome these problems, the information obtained from the following mode detection algorithm will be used to improve the trip segmentation. This is explained in Section 7.

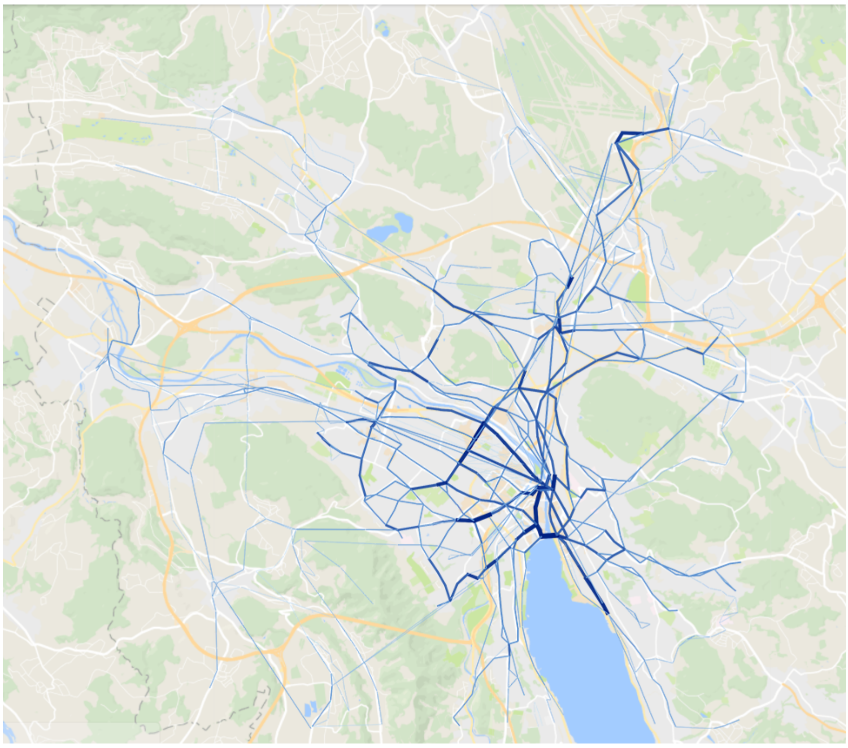
An example of these steps until the trip segmentation, can be seen in Fig. 4. The user's real path is the following: from the activity **A** the user had a short walk (**B**) to take a tram (**C**). Then the user waited for a bus in **D**, took the bus (**E**) and arrived at **F** to stay there a while. Later the user took a train (**G**), stayed at the Zürich Main Station (**H**), walked to a stop (**I**), took a tram (**J**) and walked to home (**K**). Even if the sampling frequency is different through the day, the segmentation algorithm is able to divide correctly each trip. In particular, the frequency is lower when the user is on a train (**G**) or in a tunnel (the upper-right part of **C** and **J**). It is also lower on **I**, probably because the main station has a underground part.

7 Mode Detection

Many methods have been studied to automatically collect information about the user's travel behaviour from raw GPS data. One of the major problem is the identification of travel mode. In the first papers the GPS data have been collected with the help of dedicated GPS devices (Patterson *et al.* (2003), Gong *et al.* (2012)), that are expensive for the researchers and cumbersome for the users. In the last decade, researchers started to look at smartphones as a powerful source of data for travel surveys. In fact, they are equipped with a GPS module, most of the people travel always with one of them, and they have also other sensors that can be useful for this purpose. However, the battery life is still a problem (Wu *et al.* (2016)), especially when a high sampling frequency is specified. For this reason, we developed an approach able to deal with a low sampling frequency.

One of the most recent review of transportation mode detection using GPS data collected by smartphones is presented by Nikolic and Bierlaire (2017). They stated that there are two categories of methods for mode detection: machine learning methods and hybrid methods, also relying partially on machine learning techniques or probabilistic models like Hidden Markov Models (Reddy *et al.* (2010)). Wu *et al.* (2016) made a similar systematic review and it asserted that the methodology adopted by the reviewed works is rather similar. First, some features are extracted from the sensors, then a training set is built to train a machine learning algorithm, and then the algorithm is used to predict unseen data. For instance, Zheng *et al.* (2010) proposed a supervised learning approach, using a decision tree, to infer transportation modes from GPS data. It is able to detect walking, driving, taking a bus and riding a bike. Stenneth *et al.* (2011) compared different inference models like Bayesian Net, Decision Tree, Random Forest, Naive Bayesian and Multilayer Perceptron. It is able to detect different transport means as car, bus, train, bike, walking and stationary. Furthermore, it rely not only on the GPS data, but also on transportation network data. Reddy *et al.* (2010) built its classification system using also accelerometer data in addition to GPS data and it is able to detect among stationary, walking, running, bike or motorized transport. It used a hybrid approach based on a decision tree and a Hidden Markov Model. Zhang *et al.* (2011) also presented an hybrid approach to identify travel modes: first, it used the mean speed, the maximum speed and the heading changes to identify walking, bike and motorized vehicles; then it used a classification system based on Support Vector Machines to identify if the motorized vehicle is a car, bus, tram or train. Montoya *et al.* (2015) build a system based on a Bayesian network to infer the transport mode from smartphone data (GPS, wifi, accelerometer) and transport network information like the public timetable. Finally, Patterson *et al.* (2003) presented a Bayesian model inferred in an unsupervised manner. It can distinguish between walk, drive or taking a bus and it also showed that the use of additional knowledge like bus stop location can improve the algorithm. Except Patterson *et al.* (2003), all of the mentioned studies rely on inference models that need a training set to work. For this reason

Figure 5: Public transport vehicles traffic of Zürich: darker and larger lines represent more transport vehicles travelling on a working day (26-03-2018).



a manual labelling of the transportation modes is required, that can be expensive or difficult to obtain, limiting the dimension of the dataset. In addition, most of the mode detection studies in the literature lack information about the sampling frequency of the GPS data or did not compare their algorithm with different sampling frequencies. Instead, in this study we consider this parameter crucial for a mode detection algorithm, because only with a low sampling frequency, the battery of the smartphone is not affected and then the algorithm can be used in practical applications. Therefore, for our purpose to have a system invisible for the users and that can easily work with a large number of people, we choose a data-driven approach largely different from the previous papers.

One of the most important differences is that the algorithm is completely unsupervised and it does not use any statistical inference model. In fact, it is mostly based on the realised operations data. Only Stenneth *et al.* (2011) of the cited works used this type of data, but just to build features for their inference model. Fig. 5 shows the public transport traffic network of Zürich, relative to buses, trams and trains travelling in the city. For all of these means the realized data are publicly available (SBB (2018)): more specifically, for each stop of each vehicle the scheduled and realized times of its arrival and departure are available. Since walks are detected during the trip segmentation, the mode detection is able to label a stage as a bus, tram, train or otherwise a private vehicle. Moreover, for the public transport it detects the exact vehicle that the user took.

The algorithm starts as following: all the public transport stops in a radius *R* near the starting point of the stage are detected, then all the vehicles that are stopping at one of these stops in ± *T* seconds from the starting point are selected. The same method is applied for the end point, then the intersection of the vehicles in the two groups returns a list of all the vehicles passing near the user at the begin and at the end of the stage. At this point, a likelihood function is applied to each element of the list to select the most probable vehicle the user took. Instead, if the list is empty, that means the user used a private vehicle. For the experiments in this paper the parameters have been set to *T = 250 [sec.]* and *R = max(250,min(accuracy,400)) [m]* where accuracy is a value in meters provided by the smartphone for each GPS point.

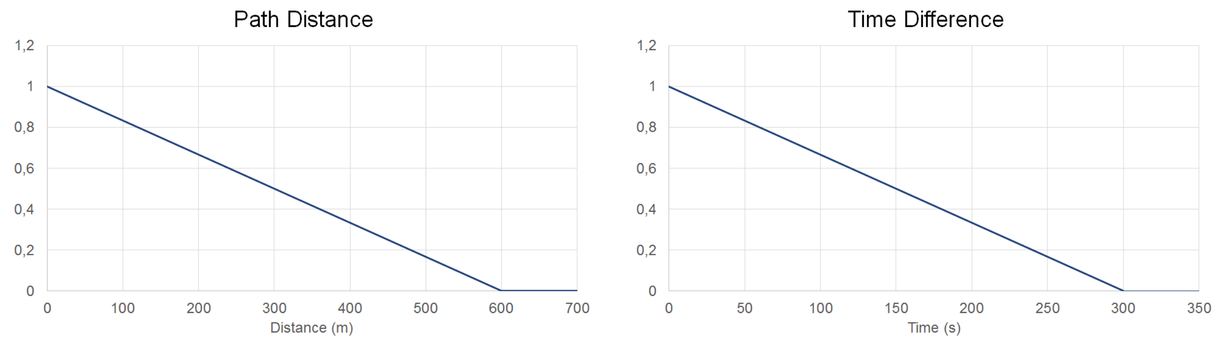
7.1 Likelihood function

The likelihood function for the mode detection is a combination of two functions comparing the paths of the user and of the vehicle, as shown below:

$$L = \lambda * TimeDifference + (1 - \lambda) * PathDistance$$

TimeDifference is the sum of the difference between the departure times and the difference between the arrival times of the user and the vehicle. *PathDistance* is the average euclidean distance between the user's points and the vehicle points. The vehicle points are calculated at the same timestamps of the user's points interpolating from the arrival time at each stop. Then, these two values are scaled like in Fig. 6 to be comparable. A value of $\lambda = 0.5$ is chosen because using only the *TimeDifference* can bring to false positive matching with vehicles that had a different path, meanwhile the *PathDistance* is not reliable if there are few user's points. In the end, the value *L* is multiplied by the times that the vehicle appeared in one of these lists of candidate vehicles of stages in the same trip. The idea behind is that if a single vehicle is a candidate to match different stages of the same trip, it is probable that the user took only one vehicle to make all the stages. Furthermore, the value of *L* is not comparable for different stages, because it is dependent from the quality of the user's path. With low quality data *L* tends to be low, instead with high quality data *L* tends to 1 for the taken means. It is important to notice, that the paths of the vehicles have always a high quality, because they are based on the position of the stops (that is fixed) and on the realized arrival time at each stop (provided by the operations and subject to fewer errors than the user's data).

Figure 6: Mode detection likelihood: The two sub-functions *PathDistance* and *TimeDifference*.



7.2 Mode detection improvements

In order to improve the mode detection algorithm and to overcome the errors obtained during the trip segmentation, three different rules are applied:

- If in a trip there is the pattern *stage-walk-stage* and the two stages are not assigned to a means, consider the three segments as a single stage and compute the mode detection.
- If in a trip there is the pattern *stage-walk-stage* and the vehicle detected for the two stages is the same, consider the three segments as a single stage performed with that vehicle.
- If a stage is not assigned to a means, look in the user's past data for a possible transfer point in the path, and try to detect two different means for the whole stage.

The first two rules are introduced in case of a wrong detection of a walk segment. For instance, this can happen if a vehicle is stuck in traffic. The last rule is introduced in case of a walk segment is not detected, that can happen if a transfer is performed quickly. In this case, the average point of the previous activities and the starting and ending points of the previous stages of the user are considered as potential changing points, if they are in a distance R (250 meters) from one point of the user's path. If some of these changing points are near to each other, the average point is considered, because they represent the same location. This represents a first attempt, never described in literature, to use the user's past data for a mode detection algorithm. In that way, only the user's relevant places are considered, useful in particular for datasets spanning multiple weeks with a huge quantity of data recorded for a single user.

7.3 Results

The user's GPS data are recorded over a period of four weeks, then it was not asked to the participants of the experiment to manually label their movements and the travel modes they took

Table 1: Dataset information: values of the different components obtained applying the whole mode detection algorithm to the dataset.

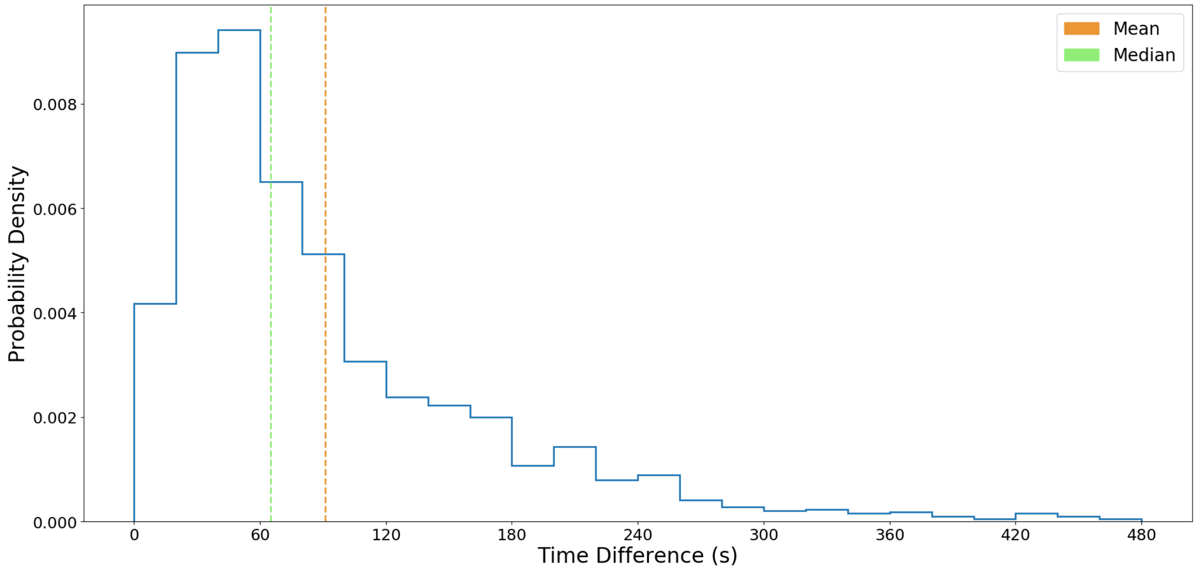
	Quantity
Activities	2931
Trips	2820
Walks	7031
Stages	4868
Detected stages	1954
Not assigned stages	522
Far stages	2392
Past Data Detected stages	128

in this period. For this reason, a ground truth for the mode detection is not available, but the quality of the algorithm can be assessed comparing the user's path with the path of the detected transport mode. Table 1 gives the results of the mode detection algorithm and the dataset, to give a better idea of its dimension. Regarding the stages, they are divided in three different groups: *detected*, *not assigned* and *far*. A stage is marked as *far* if it is performed partially or completely outside the city of Zürich (Fig. 5); it is marked as *detected* if the mode detection algorithm found a mode for it; instead, it is marked as *not assigned* if the algorithm did not find a match to a means. In this last group there are principally stages performed with a private vehicle, but also public transport stages that are not detected, because of bad GPS quality or errors during the activity identification or trip segmentation steps. Further, in the *detected* group there are also false positive detections, that can occur only in case of data with a very low quality or, for instance, if the user is always behind a bus and then the algorithm match the bus as the transport mode. Nevertheless, this case is considered rare because usually a car can overtake or has a different lane, and a bicycle has a higher travel time. The *Past Data Detected* stages, are the stages detected thanks to the use of the user's past data, as described in Section 7.2. Without using these data, the *Past Data Detected* stages would have been labelled as *not assigned*, with a increase of the *not assigned* group of the 24.5%. This value clearly shows the importance of information about the user's past travel behaviour for a mode detection algorithm.

It is important to notice that the main factor influencing the quality of the mode detection algorithm is the quality of the data, while the sampling frequency is less relevant. In fact, in the extreme case of only two points, with the exact position and time, one at the origin and the other at the destination of a user's path, the algorithm detects if it is a stage from its speed and the exact mode from the departure and arrival times of the modes in the network.

To measure the quality of the mode detection, the *TimeDifference* function (described in Sec-

Figure 7: Distribution of *TimeDifference* for all the *detected* stages in the dataset (grouped each 20 *s*)



tion 7.1) is used. In fact, if the difference between the arrival and departure times of the user and of the vehicle is low, it is reasonable to think the detection is correct. The *PathDistance* function was not chosen because a user's path can be very noisy and then not comparable with the path of the detected mode even with a correct detection. The distribution of the *TimeDifference* of all the detected stages in the dataset is shown in Fig. 7. This value, the sum of the differences between the arrival times and departure times of the user and the vehicle, is caused by two main factors: the trip segmentation and the sampling frequency. In particular, an erroneous trip segmentation can identify the beginning or the end of the stage at some points before or after the real one. Moreover, often with a low sampling frequency there are no points in the exact time the user took the means. For these reasons a mean value of 91 s and a median value of 68 s for the *TimeDifference*, shown in Fig. 7, are considered good values and a guarantee of correct matching. Instead, with higher values, such as more than 300 s, the probability of a wrong detection increase. To increase the precision of the algorithm, it is possible to decrease the values for the parameters T and R, to reduce the wrong detection, but in this way also the false negatives increase. In fact, if for instance the user is in a station, there could be a long time without signal or there are only data with low accuracy, then high values of T and R are required. In Fig. 8 for each detected stage the sampling frequency of the trip and the value of *TimeDifference* is plotted. The sampling frequency of the trip is shown, because the mode detection principally relies on the two extreme points of the stage, that are detected during the trip segmentation. Moreover, gaps of signal greater than 10 minutes are not considered for the sampling frequency as for the trip segmentation. It is possible to see that there is not a specific relationship between sampling frequency and *TimeDifference*, demonstrating that the quality of the mode detection algorithm is not dependent from the sampling frequency. In Fig. 9 the *TimeDifference*

Figure 8: Sampling frequency and *TimeDifference* for each detected stage.

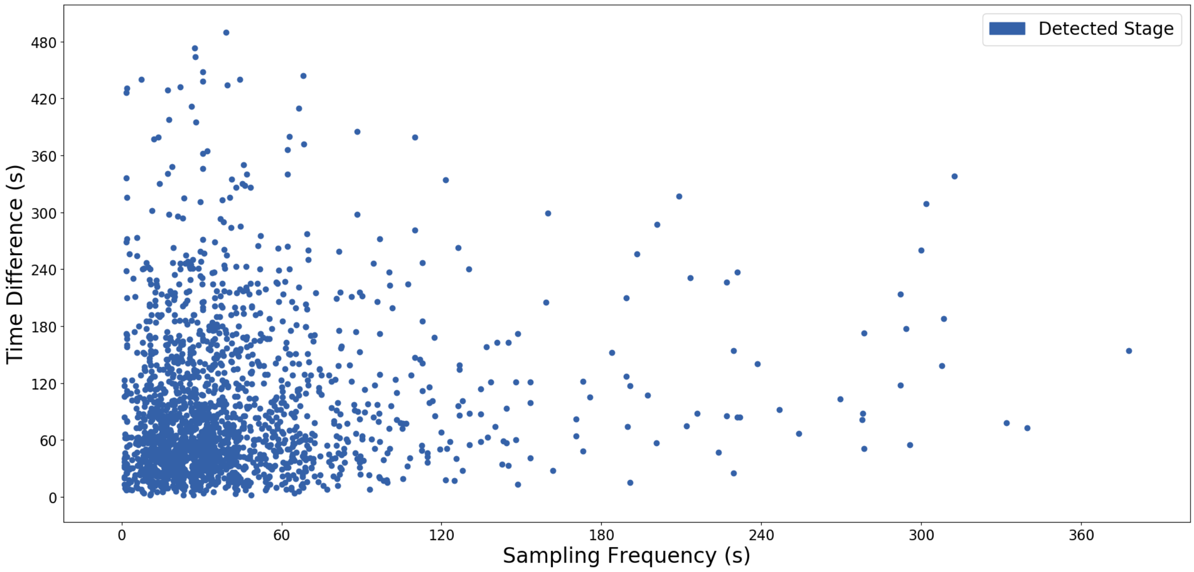
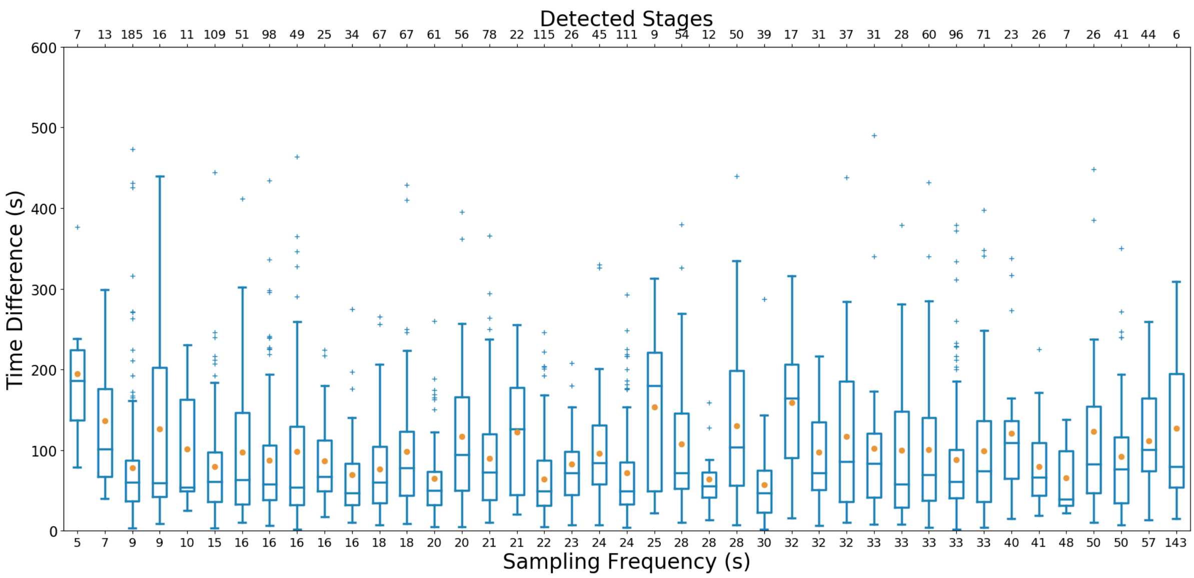


Figure 9: Values of *TimeDifference* for all the *detected* stages, grouped by user. The orange dot is the mean value. The number of detected stages and the average sampling frequency for all user's trip are shown.



is grouped for stages of the same user and the average sampling frequency for all users' trip is shown. No relationship has been found between the users' sampling frequency and the mode detection quality, demonstrating that the variation of *TimeDifference* among different users depends principally on external factors like the user's travel behaviour or his main locations, that can alter the data.

8 Discussion and Conclusions

With this work it is confirmed that GPS data collected from smartphones are a powerful means to understand users' travel information, compared to traditional travel survey methods. Moreover, we have overcome the main obstacles that arise using the smartphones for this purpose: we used a low-battery consuming application that does not affect the daily use of the smartphone; we also showed that is possible to collect users' travel information only with low-frequency and low-precision GPS data. In fact, specific algorithms for activity detection, trip segmentation and mode detection have been designed to deal with this type of data. Regarding the mode detection, our technique is totally different from what is found in the literature: in fact, most of the work is based on supervised learning, requiring a lot of efforts to manually label the data. Instead, our algorithm only relies on the combination of user's GPS data with the transportation network data, and it is also able to perform an exact vehicle imputation. Moreover, this paper represents an original attempt to use the realized data of a public transportation network for a travel survey purpose. We also introduced a way to exploit the user's past data to improve the mode detection algorithm. In fact, it is shown clearly that information about the user's past movements and main locations can be used to better understand the user's future behaviour. For this reason, in a future work we would like to take advantage of the user's travel history to identify possible mobility patterns. We will also include in the mode detection a way to distinguish private vehicles between cars and bicycles, based on speed patterns and the user's past data. With this work we demonstrated the feasibility of build travel surveys with low costs and without any negative impact on the users, opening the way for a future study based on a large scale dataset, that will lead to a different manner to conduct travel surveys. We will be able to extract different patterns and to shed some light on understanding choices of travellers that can not be derived from a small dataset. In that way, it will be possible to analyse why people make some travel choices and what are their main criteria for these choices. Furthermore, a future goal will also be to understand how passengers react to disturbances in the transport network and to the mitigation action undertaken by operators, in order to design good and effective mitigation actions.

Acknowledgement

This work was partly supported by projects funded by the Swiss Federal Road Office (project ASTRA 2017/006) and the Swiss Innovation Agency (Innosuisse, SCCER Mobility).
We thanks all the participants of the study, Swiss Federal Railways (SBB) for the open data, OpenStreetMap and Google Maps for the shown maps.

9 References

Ansari Lari, Z. and A. Golroo (2015) Automated Transportation Mode Detection Using Smart Phone Applications via Machine Learning: Case Study Mega City of Tehran, paper presented at the *94th Annual Meeting of the Transportation Research Board, Washington*.

Biljecki, F., H. Ledoux and P. van Oosterom (2013) Transportation Mode-based Segmentation and Classification of Movement Trajectories, *International Journal of Geographical Information Science*, **27** (2) 385–407.

Bohte, W. and K. Maat (2009) Deriving and validating trip purposes and travel modes for multi-day GPS-based travel surveys: A large-scale application in the Netherlands, *Transportation Research Part C: Emerging Technologies*, **17** (3) 285–297.

Bricka, S. and C. Bhat (2006) Comparative Analysis of Global Positioning System-Based and Travel Survey-Based Data, *Transportation Research Record: Journal of the Transportation Research Board*, **1972**, 9–20.

Chung, E.-H. and A. Shalaby (2005) A Trip Reconstruction Tool for GPS-based Personal Travel Surveys, *Transportation Planning and Technology*, **28**, 381–401.

Cottrill, C., F. Pereira, F. Zhao, I. Dias, H. Lim, M. Ben-Akiva and P. Zegras (2013) Future Mobility Survey: Experience in developing a smartphone-based travel survey in singapore, *Transportation Research Record: Journal of the Transportation Research Board*, **2354**, 59–67.

Gong, H., C. Chen, E. Bialostozky and C. T. Lawson (2012) A GPS/GIS method for travel mode detection in New York City, *Computers, Environment and Urban Systems*, **36**, 131 – 139.

Jones, P. and M. Clarke (1988) The significance and measurement of variability in travel behaviour, *Transportation*, **15** (1-2) 65–87.

Jun, J., R. Guensler and J. Ogle (2006) Smoothing Methods to Minimize Impact of Global Positioning System Random Error on Travel Distance, Speed, and Acceleration Profile Estimates, *Transportation Research Record: Journal of the Transportation Research Board*, **1972**, 141–150.

Liao, L., D. J. Patterson, D. Fox and H. Kautz (2006) Building personal maps from GPS data, *Annals of the New York Academy of Sciences*, **1093**, 249–265.

Montini, L., N. Rieser-Schüssler, A. Horni and K. W. Axhausen (2014) Trip Purpose Identification from GPS Tracks, *Transportation Research Record: Journal of the Transportation Research Board*, **2405**, 16–23.

Montoya, D., S. Abiteboul and P. Senellart (2015) Hup-me: inferring and reconciling a timeline of user activity from rich smartphone data, paper presented at the *ACM SIGSPATIAL, International Conference on Advances in Geographic Information Systems*, 1–4.

Nikolic, M. and M. Bierlaire (2017) Review of transportation mode detection approaches based on smartphone data, paper presented at the *17th Swiss Transport Research Conference, Ascona*.

Nitsche, P., P. Widhalm, S. Breuss, N. Brändle and P. Maurer (2014) Supporting large-scale travel surveys with smartphones - a practical approach, *Transportation Research Part C: Emerging Technologies*, **43**, 212–221.

Ogle, J., R. Guensler, W. Bachman, M. Koutsak and J. Wolf (2002) Accuracy of Global Positioning System for Determining Driver Performance Parameters, *Transportation Research Record: Journal of the Transportation Research Board*, **1818**, 12–24.

Oliveira, M., P. Vovsha, J. Wolf, Y. Birotker, D. Givon and J. Paasche (2011) Global Positioning System-Assisted Prompted Recall Household Travel Survey to Support Development of Advanced Travel Model in Jerusalem, Israel, *Transportation Research Record: Journal of the Transportation Research Board*, **2246**, 16–23.

Pas, E. I. and S. Sundar (1995) Intrapersonal variability in daily urban travel behavior: Some additional evidence, *Transportation*, **22** (2) 135–150.

Patterson, D. J., L. Liao, D. Fox and H. Kautz (2003) Inferring High-Level Behavior from Low-Level Sensors, paper presented at the *UbiComp 2003: Ubiquitous Computing*.

Pearson, D. (2001) Global Positioning System (GPS) and travel surveys: Results from the 1997 Austin household survey, paper presented at the *8th Conference on the Application of Transportation Planning Methods, Corpus Christi*.

Pelletier, M.-P., M. Trépanier and C. Morency (2011) Smart card data use in public transit: A literature review, *Transportation Research Part C: Emerging Technologies*, **19** (4) 557–568.

Reddy, S., M. Mun, J. Burke, D. Estrin, M. Hansen and M. Srivastava (2010) Using Mobile Phones to Determine Transportation Modes, *ACM Trans. Sen. Netw.*, **6** (2) 13:1–13:27.

SBB (2018) SBB opendata, `https://opentransportdata.swiss`.

Schlich, R. and K. W. Axhausen (2005) Analysing Interpersonal Variability for Homogeneous Groups of Travellers, *Arbeitsberichte Verkehrs- und Raumplanung*, **296**.

Schuessler, N. and K. W. Axhausen (2009) Processing Raw Data from Global Positioning Systems Without Additional Information, *Transportation Research Record: Journal of the Transportation Research Board*, **2105**, 28–36.

Shin, D., D. Aliaga, B. Tunçer, S. M. Arisona, S. Kim, D. Zünd and G. Schmitt (2015) Urban sensing: Using smartphones for transportation mode classification, *Computers, Environment and Urban Systems*, **53**, 76–86.

Stenneth, L., O. Wolfson, P. S. Yu and B. Xu (2011) Transportation Mode Detection Using Mobile Phones and GIS Information, paper presented at the *19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*.

Stopher, P., Q. Jiang and C. FitzGerald (2005) Processing GPS data from travel surveys, *28th Australasian Transport Research Forum, ATRF 05, Sydney, New South Wales, Australia*.

Stopher, P., K. Kockelman, S. Greaves and E. Clifford (2008) Reducing Burden and Sample Sizes in Multiday Household Travel Surveys, *Transportation Research Record: Journal of the Transportation Research Board*, **2064**, 12–18.

Susilo, Y. and K. W. Axhausen (2007) How firm are you? A study of the stability of individual activity-travel-location pattern using Herfindahl Index, paper presented at the *11th World Conference of Tranport Research*, Berkely, CA, USA.

Susilo, Y. and R. Kitamura (2005) Analysis of Day-to-Day Variability in an Individual's Action Space: Exploration of 6-Week Mobidrive Travel Diary Data, *Transportation Research Record: Journal of the Transportation Research Board*, **1902**, 124–133.

Tsui, S. and A. Shalaby (2006) Enhanced System for Link and Mode Identification for Personal Travel Surveys Based on Global Positioning Systems, *Transportation Research Record: Journal of the Transportation Research Board*, **1972**, 38–45.

Wolf, J., S. Schoenfelder, U. Samaga, M. Oliveira and K. W. Axhausen (2004) Eighty Weeks of Global Positioning System Traces: Approaches to Enriching Trip Information, *Transportation Research Record: Journal of the Transportation Research Board*, **1870**, 46–54.

Wu, L., B. Yang and P. Jing (2016) Travel Mode Detection Based on GPS Raw Data Collected by Smartphones: A Systematic Review of the Existing Methodologies, *Information*, **7** (4) 67.

Zhang, L., S. Dalyot, D. Eggert and M. Sester (2011) Multi-Stage Approach to Travel-Mode Segmentation and Classification of GPS Traces, *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, **3825**, 87–93.

Zheng, Y., Y. Chen, Q. Li, X. Xie and W.-Y. Ma (2010) Understanding Transportation Modes Based on GPS Data for Web Applications, *ACM Trans. Web*, **4** (1) 1:1–1:36.

Zhu, Q., M. Zhu, M. Li, M. Fu, Z. Huang, Q. Gan and Z. Zhou (2016) Identifying Transportation Modes from Raw GPS Data, paper presented at the *Social Computing*, Communications in Computer and Information Science, 395–409.