universität wien

MASTERARBEIT | MASTER'S THESIS

Titel | Title

Protecting sensitive location information for mobile fitness applications with spatial k-anonymity

verfasst von | submitted by

Robby Heusequin BSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2024

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 856
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Kartographie und Geoinformation
Betreut von | Supervisor:	Ass.-Prof. Dr. Ourania Kounadi BSc MSc

Contents

List of Figures

List of Tables

Kurzfassung

Diese Masterarbeit untersucht die aktuellen Methoden zum Schutz der privaten Daten, die in mobilen Fitness-Applikationen eingesetzt werden. Das Teilen von Trainingsaktivitäten in Fitness-Anwendungen kann dazu führen, dass die sensiblen Standortdaten von Personen an andere Sportler weitergegeben werden. Es wurden Geomaskierungsmethoden entwickelt, um sensible (Standort-)Informationen von Einzelpersonen zu schützen. Die Implementierung von solchen Datenschutzmethoden in diesen Anwendungen zielt darauf ab, sensible Standortinformationen zu schützen. Allerdings zeigt sich in der Praxis, dass dieser Ansatz nicht immer effektiv ist. Das Ziel dieser Studie war es, eine neue Methode zum Schutz der sensiblen Standortinformationen zu entwickeln, die auf räumlicher k-Anonymität für mobile Fitness-Applikationen basiert. Dadurch sollen die sensiblen Standortinformationen erfolgreich vor den anderen Benutzern verborgen werden. Die empirische Untersuchung wurde an simulierten Trajektorien in vier Wiener Bezirken mit zwei verschiedenen räumlichen k-Anonymitätsstufen durchgeführt. Die Resultate wurden visuell untersucht und mit den ursprünglichen Routen verglichen. Im Anschluss wurden die mittlere und die mediane Distanzverschiebung sowie die Differenz der insgesamt zurückgelegten Strecke untersucht. Die wesentlichen Erkenntnisse dieser Arbeit sind, dass die entwickelte Geomasking-Methode eine effektive Möglichkeit darstellt, die sensiblen Standortinformationen der Nutzer zu schützen. Es wurden jedoch einige Schwächen festgestellt, die die Anwendung der entwickelten Geomaskierungsmethode für Nutzer beeinflussen können. Abschließend werden weitere Empfehlungen für Verbesserungen der entwickelten Methode zum Schutz der sensiblen Standortinformationen vorgestellt, um die während der Testphase in dieser Arbeit festgestellten Schwächen zu beheben.

Abstract

This master's thesis examines the current privacy protection methods used in mobile fitness tracking applications. Sharing training activities in fitness tracking applications can lead to the individual's sensitive location information being shared with other athletes. Geomasking methods have been developed to protect sensitive (location) information of individuals. The implementation of privacy protection methods in these applications aim to protect sensitive location information but are not always effective. The goal of this study was to develop a new privacy protection method that is based on spatial k-anonymity for mobile fitness tracking applications to successfully mask the sensitive location information from the users (for example their home location). Research was conducted on simulated trajectories in four districts in Vienna with two different spatial k-anonymity levels. Furthermore, the results were visually explored and compared with the original routes. Then, the mean and median distance displacement and the difference in the total distance travelled between the original and masked route was examined. The main findings of this thesis are that the developed geomasking method performs well for protecting the sensitive location information from the athletes. However, some weaknesses were found, which can influence the use of the developed geomasking method for athletes. Lastly, further recommendations for enhancements to the developed privacy protection method are presented to address the shortcomings identified during the testing phase in this thesis.

Acknowledgements

I would like to express my gratitude to all those who have provided me with invaluable support and assistance throughout the course of my dissertation. First of all, I want to thank my supervisor, Ass-Prof. Dr. Ourania Kounadi, BSc MSc, for all the guidance, support and interesting discussions throughout the entire research journey. I would also like to express my gratitude and love to my fiancé and family who have always supported and encouraged me. And finally, I'd like to thank all my friends and colleagues I've met during my studies.

1 Introduction

This first chapter acts as an introduction to the master thesis. First of all, the theoretical background of the subject of the research is presented. Proceeding to the second section, the focus shifts to the difficulties surrounding this subject. Then, the objectives of this thesis are outlined, together with the research questions. Also, in this chapter the research boundaries are established. Furthermore, the methodological approach of this thesis is described. The last section presents a summary of the thesis structure.

1.1 Theoretical Background

In the era of big data, safeguarding personal privacy emerges as a serious concern. Those who contribute data through the use of mobile devices face the potential of identification and infringement upon their privacy, if data handling practices are inadequate. Due to the rapid advancement of technology, the collection, storage and analysis of large data quantities is an easier task than ever before (Wang & Kwan, 2020). Consequently, this has made it complicated to define what privacy is, as it constantly changes and is also context dependent (Zhang & McKenzie, 2023).

This is why location privacy, also called geoprivacy, is extremely important. Individuals should have the control to determine the conditions under which their gathered location information is disclosed to third parties, including when, why and for what purposes such data may be released (Cremonini et al., 2013).

Smart devices, such as smartphones, smart- and fitness wearables connect us to the world, but also actively collect an immense amount of personal and geospatial data from the users through mobile applications (Keßler & McKenzie, 2018). The collected data gets stored and shared with other applications and/or companies. This happens through application programming interfaces (API), but developers also share this personal data with third parties, where this information gets used for commercial and research purposes, such as personalized advertisement (Grundy et al., 2017). Therefore, it is important, that users understand and know what happens with their personal and location data.

Furthermore, many users have multiple social network accounts, which means that they share different kind of information on these platforms. For example, someone has a Facebook account to chat with friends and share some of their activities. This same person also has an Instagram account, where pictures and videos are shared with their followers. Additionally, this person uses a fitness tracking application for tracking some personal health records, such as calories intake, movement, training sessions, sleep etc. All this different sensitive personal and geospatial information gets stored and can be used by third parties and companies to connect this data and produce aggregated user profiles (Li, 2015).

Aggregated user profiles are a huge privacy risk and in this thesis **a method is developed that could help with protecting a user's sensitive location information.**

Individuals, who want to publish maps, which could potentially contain sensitive location information and want to ensure the protection of the sensitive data involved, need to have knowledge about the sensitivity of the spatial data involved and also require an expertise in geoinformation systems to effectively use a GIS workflow (Swanlund et al., 2020a). Most efforts to safeguard location information in geoprivacy research has mainly focused on so called geomasking techniques, which purposely adds inaccuracy or obfuscation to the geographical data that has been collected (Seidl et al., 2020). As shown in Kounadi and Leitner (2014) quite a few masking techniques have already been established and each has their own advantages and disadvantages. However, even when geographical masking techniques are used to protect location information, there is still a risk of re-identification (Hassan et al., 2018; Kao et al., 2017; Kounadi & Leitner, 2014). Another aspect to consider regarding location privacy concerns the nature of user data shared on social media platforms and the extent of their awareness of the potentially sensitive information they disclose (Alrayes et al., 2020).

1.2 Statement of The Problem

In the last decade fitness tracking applications are becoming more popular, giving the users a possibility to track their fitness activities, such as running, cycling and other sport activities. These fitness applications can also share their results with followers and friends on their platform, but also on other social network platforms (Hassan et al., 2018). As mentioned previously, sharing this personal and geospatial information has its privacy risks.

A great example is when the Global Positioning System (GPS) tracking company Strava published a heat map with locations and movements of users, who used the company's tracking services from 2015 until September 2017. In this map Europe and the United States were very bright, as there was a lot of activity in these parts of the world, but other parts of the world were dark with a few exceptions. In areas of conflict or deserts there were some bright spots and these locations were either U.S. military bases or other unknown and potentially sensitive locations, which are extremely sensitive data for authorities (Liz, 2018).

Furthermore, there is the risk for users, who share their training routes with followers, friends and other users. When a training session is completed, the user gets a summary of some statistics (e.g., distance travelled) and this summary can be expanded further, when a fitness wearable (e.g., hearth rate, sleep etc.) is connected with the application. The recap can also include a map, showing the route that the users ran or cycled. These summaries can be published to the user profile and on other social media platforms to compete with other users or to simply share the

training routes. Sharing this information has privacy risks, as sensitive geospatial information can be included such as home address, workplace etc. (Hassan et al., 2018).

However, some of these fitness tracking applications provide some privacy mechanisms, that should protect this sensitive data. Hassan et al. (2018) described four mechanisms that some fitness applications provide and tested one mechanism, called endpoint privacy zone (EPZ), if it protects the sensitive location information. Users often start their training session at a sensitive location and most applications give the users the option to hide parts of their route, which are within a certain distance of the sensitive location. Hassan et al. (2018) wrote an algorithm to attack the protected location information from users on the fitness tracking application Strava. They ran their algorithm on a dataset with over 2 million EPZ-enabled training activities from around 430 thousand athletes and using their technique they were able to identify 84% protected locations of users with more than one EPZ-enabled activity. When a user recorded at least three EPZ-enabled activities the accuracy of the algorithm went up to 95,1%. They conclude that the EPZ privacy mechanism is ineffective for hiding a user's sensitive location information as proven by their re-identification results.

1.3 Objectives and Research Questions

The main objective of this master thesis is to develop a method to protect sensitive location information (home address, work address etc.) from users who use fitness applications to track their training sessions. Sub-objectives are to evaluate the current protective mechanisms and protection needs of these applications as well as the potential effectiveness of existing location protection techniques. To achieve these objectives, the following research questions are posed:

1. Which protective mechanisms do mobile fitness applications offer?
2. What are the limitations of the existing protective mechanisms by such apps in terms of usage and/or risk of re-identification?
3. Can existing geographical masking methods, originally designed for discrete location data, be applied to spatio-temporal trajectories of individuals to protect their sensitive locations?
4. How can the privacy measure of spatial k-anonymity (ska) be applied to individual trajectories and prevent inference attacks of the individual sensitive locations?

1.4 Methodological Approach

In this Master thesis the theoretical part is examined with the help of a literature review. Here, different types of geographical masking methods for the protection of personal data and geospatial data are explained and examined on how well they are able to protect a user's personal data. Furthermore, the potential for re-identification of these methods is investigated and described in detail. Additionally, the spatial k-anonymity method is explained in depth as a means of ensuring data privacy.

For the practical part a script is programmed with Python, that uses a ska-based approach to protect a user's sensitive location information for mobile fitness applications. In the literature most protective mechanisms use distance to displace a location and for this thesis a ska-based method using the nearest neighbour algorithm with addresses and street intersections is developed and tested. The data that is used for developing and testing the method are simulated scenarios of individual trajectories from an open-source API, called Openrouteservice. Lastly, results will then be presented, examined and discussed.

1.5 Structure of the Thesis

This section presents an overview of the thesis structure. Following an introduction into the topic of the dissertation, a literature review is presented in Chapter 2. First of all, a brief overview of the history of (location) privacy is presented in section 2.1. Furthermore, in section 2.2 the different types of spatial and personal data that are collected from mobile applications are explored and illustrated, multiple established geomasking techniques are presented and the current privacy protection mechanisms offered by fitness tracking applications are introduced and described. The methodology part of this thesis is provided in Chapter 3. First, the software utilised in this thesis is introduced in section 3.1. The defined study area is described in section 3.2 and the data used for the analysis is presented in section 3.3. Finally, the geomasking method developed during this thesis is provided in section 3.4. In Chapter 4 the results are presented and discussed. The sections provide a visual exploration of the outcomes from the developed geomasking technique in the different study areas. This is followed by an examination of the mean and median distance displacement, and finally, a recommendation on how to handle the travelled distance, which is altered during the use of the privacy protection method.

The last chapter summarizes the findings and answers the research questions of the thesis. Moreover, the limitations of the work are presented and recommendations for future research on the developed geomasking method is given. The Python Code from the developed privacy protection method is included in Appendix B.

2 Literature Review

The second chapter of this thesis presents the theoretical framework for the research topic, which is a literature review. First, a brief overview about the history of (location) privacy will be reviewed. Next, two high-profile privacy incidents will be briefly described, which shows why privacy legislation and regulations are important to protect personal information. Then, there will be a demonstration and explanation of the different types of spatial and personal data that can be collected from mobile applications. Furthermore, some established geographic masking methods will be briefly explained and demonstrated. Then, privacy protection methods provided by fitness tracking applications will be explored and their effectiveness will be evaluated on the basis of existing literature. Finally, the protection metrics of k-anonymity and spatial k-anonymity will be explained, which can be used to measure the re-identification possibility of individuals.

2.1 Framing Privacy and Location Privacy

This section provides a brief overview of the evolution of location privacy, from before the digital age to the current sophisticated technologies that are at our disposal. Due to this development and the increased collection of data, some privacy incidents have occurred. As a result, privacy legislation and regulations have been introduced.

2.1.1 The Pre-Digital Era

Before the digital revolution, concerns over location privacy have existed for centuries and were limited to physical surveillance, espionage, and tracking by law enforcement agencies or government agencies. Electronic surveillance was conducted by means of spying devices like miniature radio transmitters, allowing for the observation of a person's location and other personal details without their consent or knowledge. Also, small cameras were secretly placed within rooms for the purpose of recording personal conversations among individuals, thereby enabling identification of their whereabouts (Westin, 1966).

These were not the sole forms of surveillance methods used before the digital era, but they served as the forerunners in gathering personal and location data.

It is also worth noting that, before the existence of GPS devices, acquiring location data was especially difficult. Therefore, there were limited general concerns due to the lack of easily available and extensively used location data (Stopher et al., 2006).

2.1.2 The Emergence of GPS Devices (Late 20th Century)

In the late 20th century, GPS devices made their market debut, but their use was restricted to cars only due to the absence of an inbuilt battery and reliance on the car's power supply. Early devices were known as active GPS devices, which means that the user had to enter the location data as he or she was driving around. Around the same time, the first wearable device was developed for use while cycling. The device had its own internal battery and weighed two kilograms, which made it impractical to carry while riding a bike (Stopher et al., 2006).

In the coming years development started for passive GPS devices. These devices don't need the user to interact with the GPS device anymore and passively collects the location data. Again, these devices were limited to the use in cars and functioned when the car was running. When the car was turned off no data was being collected as it did not have any power. A wearable device with a receiver also was developed that could be carried in a bag and was powered by batteries (Stopher et al., 2006).

During this early phase of GPS devices, the privacy concerns were limited because the devices were big and heavy and were mainly used in cars.

2.1.3 Mobile Devices and Location-Based Services

Since the introduction of mobile devices in our society, concerns about personal data privacy have been raised (Martin & Nissenbaum, 2020; Patrick et al., 2008). Over the years, the use and sale of mobile devices has grown rapidly in our digital society (Statista, 2023), and as a result, location-based services (LBS) are now being utilised more frequently through mobile applications. Therefore, an increasing amount of location data is being gathered (Bellavista et al., 2008; Huang et al., 2018).

Because of the evolution of LBS-capable mobile devices users started to develop applications that transmit location data to servers and share it with other people. These innovations transformed into successful businesses and were the predecessors for location sharing applications we know today (Bellavista et al., 2008).

With the constant development of mobile devices, the technology behind LBS has also been evolving quickly (Huang et al., 2018). Raper et al. (2007) showed that most mobile applications with LBS were navigation applications and mobile guides. A navigation application helps the users go from place A to B, either by foot, by bike or by car. Mobile guide applications, however, were often offered as a separate device, that could be rented for tour guides etc. Although this has changed, and such applications are now available for download on personal devices and are mainly used for recreational purposes and tourism (Huang et al., 2018).

In recent years other mobile applications that use LBS have been developed and are being used more in the daily lives of our society. In Figure 1, some of the different types of LBS applications are being shown.

Figure 1: Example of LBS applications. Source: Huang et al. (2018)

However, the rise of mobile device usage and LBS applications has led to an increase in the collection and storing of location data. This has led to some privacy incidents being reported by known online newspapers (Hinds et al., 2020; Liz, 2018) and therefore privacy concerns from mobile application users have risen. That's why privacy legislations and regulations started to be introduced by some regions over the world (Georgiadou et al., 2019; Martin & Nissenbaum, 2020).

2.1.4 High-Profile Privacy Incidents

One privacy incident was already briefly mentioned in the first chapter of this thesis, where Strava posted a worldwide map of the tracking activities from users between 2015 and 2017. People started analysing this published map and some secret military U.S bases were found, because soldiers used their fitness tracking application (Liz, 2018).

Another high-profile privacy incident was the Cambridge Analytica case. Here, personal and private information of millions of Facebook accounts was gathered through quiz. Not only the data from the person participating in the quiz was collected, but also the data from their Facebook friends were acquired. The acquired data got shared with Cambridge Analytica,

which in turn used this data together with an algorithm to target these users with personalised advertisement (ur Rehman, 2019).

As these two examples show, it is important that there are strict privacy legislation and regulations in place that should protect sensitive personal information and prevent these incidents from happening.

2.1.5 Privacy Legislation and Regulation

As briefly mentioned before, concerns about geoprivacy especially since these technology advancements are not new. In 2014, there was a report of the President's Council of Advisors on Science and Technology about big data. Here they focused on how the private and public sectors could maximize the benefits of big data, but also how to minimize the risks involved. Location privacy issues were one of the main harms that got recognized in our data-driven world by this report (Agnellutti, 2014).

In May 2018, the European Union introduced the General Data Protection Regulation (GDPR), which protects the personal data of natural persons. This regulation restricts how companies are allowed to acquire, store and share personal data in and outside of the EU with third parties (Georgiadou et al., 2019; Seidl et al., 2020). The GDPR is a first step in the right direction for better privacy protection, although there is still a lack of general information for the designers and developers of applications on how to comply with these regulations (Ataei et al., 2018).

Due to the establishment of the GDPR in the EU, other regions such as the United States also started changing. The state of California introduced the California Consumer Privacy Act (CCPA) in 2018 and this became effective on January 1, 2020. Historically, the United States didn't have any privacy regulations that protected consumers from the acquisition, sharing and selling of their personal information. Privacy laws were always on state-level, which means that every state could define their privacy regulations (Baik, 2020; Goldman, 2020).

Another method to protect personal and location information is using location privacy-preserving mechanisms (LPPMs), which separates the users identity from their personal and location data (Alrayes & Abdelmoty, 2017). Over the last two decades many methods have been developed and explored by researchers and some of these techniques will be explained and explored in the next chapter of this thesis.

2.2 Understanding and Securing Personal and Spatial Data

Personal data can be collected, stored and shared in various different ways. The following chapter will go more in detail about what this kind of data is and what can identify a natural person. Furthermore, different types of datasets in the context of location privacy will be explored and shown. This is followed by a more detailed description and demonstration of already established geomasking techniques. Moreover, available privacy protection methods in fitness tracking applications will be explored and explained. Lastly, the protection metric k-anonymity and spatial k-anonymity will be explored further.

2.2.1 Defining Personal and Spatial Data

Personal data refers to information relating to a natural person, by which they can be identified. There are three distinct types of such data: observed, volunteered and inferred. Observed data can be acquired by monitoring a person's activity. Volunteered data is data that the person "willingly" gave up to the data collector and lastly, inferred data is captured without the knowledge of the natural person, which involves conclusions (i.e. additional information) drawn from further processing of observed and volunteered data (Georgiadou et al., 2019).

These three different types of data include not only non-spatial data but also spatial data. Spatial data, otherwise, also known as location data, is data that consists of information that can lead to revealing the position of a natural person. This data can be collected in diverse ways like GPS coordinates, postal codes, maps and street addresses (Agnellutti, 2014; Georgiadou et al., 2019; Martin & Nissenbaum, 2020).

Liu et al. (2018) broke down personal data with location information from a data privacy perspective into three attributes. These three attributes are identity, spatial information and time, as seen in Figure 2.

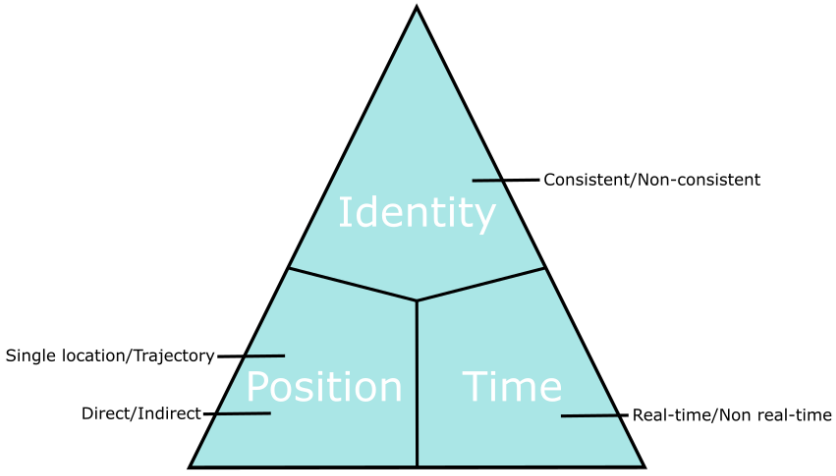


Figure 2: Personal data with location information. Adapted from Liu et al. (2018)

First, we look at the data that can disclose user's identity, other private information, or uniquely identify a user. Georgiadou et al. (2019) categorize the attributes of personal data as:

- **Unique identifier**: this falls under the broad category of identifiers and can be linked to a single entity or a data subject.

- **Key identifier**: a data element that can easily identify a natural person. Such unique elements are for example the person's name, email address, social media names, phone number, social security number, passport number etc.

- **Quasi-identifier**: a data element that does not allow for an immediate identification of a natural person, but combining quasi-identifiers, could potentially lead to the identification of the natural person. Some examples of quasi-identifiers are gender, birth data, zip code and medical conditions.

- **Private attribute**: data elements that could be private for a natural person, but even adding multiple private attributes will not lead to an identification of a person. A private attribute could be favourite food or drink, political preference, health outcome, sexual orientation etc.

SSN	Name	Surname	Email	Gender	Birthdate	ZIP Code
1111010680	Max	Mustermann	max.mustermann@example.com	Male	01.06.1980	1010
3333050901	Louisa	Gruber	louisa.gruber@example.com	Female	05.09.2001	2010
2222200295	Patrick	Chan	patrick.chan@example.com	Male	20.02.1995	3010
4444280289	Lisa	Müller	lisa.mueller@example.com	Female	28.02.1989	4010
5555251273	Valerie	Moss	valerie.moss@example.com	Female	25.12.1973	5010

Unique identifiers | Key identifiers | Quasi-identifiers

Table 1: Examples of unique, key and quasi-identifiers.

Next, with location data and time involved, the structure of these datasets can be different and much more complex (Georgiadou et al., 2019; Liu et al., 2018):

- **Location**: data, that can either be a set of coordinates, such as longitude and latitude or other information that can be linked to a certain location.

- **Location with co-variates**: a dataset with spatial data, as described in the first point, but this dataset is data richer and also includes quasi-identifiers and/or private attributes.

- **Timed location**: a dataset, which includes both location information and temporal information, resulting in a trajectory, which consists of data points with timestamps. Time can also be split into live data and non-real time data.

- **Timed location with co-variates**: this final dataset includes all three attributes. It contains quasi-identifiers, private attributes and spatiotemporal data (location and temporal information).

User ID	X	Y	Timestamp
91356	48,2087	16,3724	20.06.2023 09:00
91356	48,2109	16,3754	20.06.2023 09:05
91356	48,2135	16,3776	20.06.2023 09:10
91356	48,2137	16,3811	20.06.2023 09:15
91356	48,2148	16,3847	20.06.2023 09:20
91356	48,2177	16,3902	20.06.2023 09:25
91356	48,2189	16,3951	20.06.2023 09:30
	Location		Time

Table 2: Example of a timed location dataset

After examining various types of (spatial) datasets, the following section explains and presents multiple geomasking methods.

2.2.2 Location Privacy Protection Mechanisms (LPPMs)

Armstrong et al. (1999) defined location privacy protection mechanisms, also known as geographic masks, as techniques that modify the location information of health records to protect the privacy of the individuals' and enable a geographical data analysis. The first geographic masks that were developed primarily aggregated the spatial data into larger geographical areas. This works well for preserving the privacy. However, this method has some disadvantages for a further analysis of the data. The analysis is exposed to the modifiable areal unit problem and affects the detection of clusters in the dataset.

Over the course of two decades researchers have proposed and developed multiple geographic masking techniques. Affine transformations are one of the more basic techniques, where a geographic point can be displaced by translation, rotation, and scale. They can also be used in combination to increase the effectiveness of this masking method (Armstrong et al., 1999). This masking technique has been used in the past with some slight variations, called the flipping methodology, by Leitner and Curtis (2004). The horizontal, vertical, or both axes of the map were inverted. Generally affine transformations have not been widely used by researchers (Swanlund et al., 2020a). An advantage of this geomasking method is that the location

information is well preserved for further data analysis. However, this is also the main disadvantage, as an attacker, just needs to identify two data points correctly to re-identify the remaining data points in the dataset (Armstrong et al., 1999).

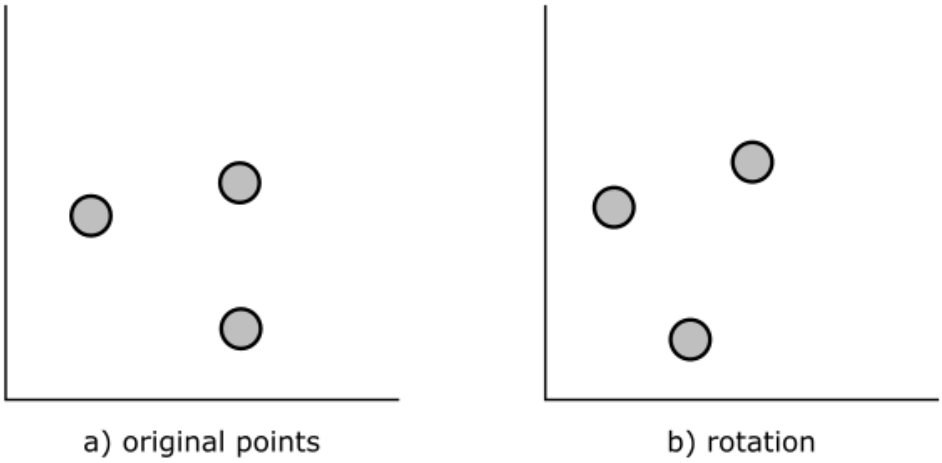


Figure 3: Affine transformation – rotation. Adapted from Armstrong et al. (1999)

Another masking method proposed by Armstrong et al. (1999), which has been very popular and has been often used by researchers is random perturbation. This technique individually moves each data point by a specific angle and a random distance in a specified area around the original point. Technically speaking, this method can be easily implemented in a GIS system by first drawing a buffer with the same radius around every point, then randomly moving the point within this buffer area and finally deleting the initial points and buffers, so that the masked points are the only points left. If researchers use this masking method with a large enough buffer area, random perturbation has a better chance of protecting privacy than affine transformation. Since, every point in the dataset is displaced randomly, which in turn makes it harder to re-identify the entire dataset, as each point has been moved independently of one another (Swanlund et al., 2020a). Nevertheless, one disadvantage is that a point that has been geomasked may be moved to the same or a nearby location as the original point (Hampton et al., 2010). Another problem is that the masked point can be moved into a body of water or an inhabited area, as this is not taken into account (Zhang et al., 2017). Furthermore, this geomasking technique has been adapted to use the population density, so that the maximum distance is actively adapted. This is necessary because some datasets include urban and rural areas and the maximum displacement distance needs to be bigger in rural areas than in urban areas to provide a sufficient privacy protection (Swanlund et al., 2020a).

An illustration of the geomasking method of random perturbation can be observed in Figure 4, wherein the point can be displaced by a distance between 0 and R2.

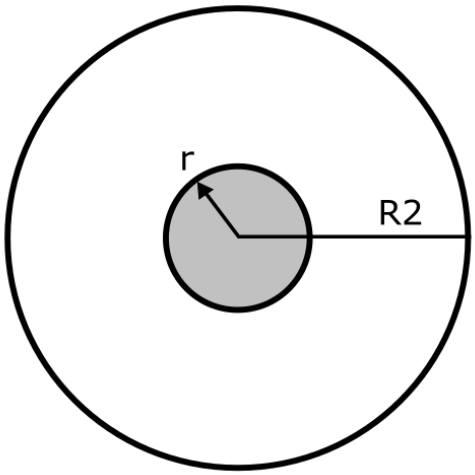


Figure 4: Random Perturbation. Adapted from Hampton et al. (2010)

One further geomasking method is donut geomasking. This masking technique is nearly identical to random perturbation but also includes a minimum distance, with which the point has to be displaced. Therefore, the possible behaviour of random perturbation, where a point could be displaced by a short distance, can be negated. Although an attacker may not be aware that a data point has only been moved in close proximity to its original location, this still represents a weakness, as it allows for partial re-identification of certain points within the dataset (Swanlund et al., 2020a). Following this small addition Hampton et al. (2010) found that the privacy protection with the donut geomasking technique is generally better than random perturbation. However, as for random perturbation, a masked point could still be displaced into inhabited areas, such as a body of water (Zhang et al., 2017). As seen in Figure 5, the technique has its name from the shape it forms on a map. The red point represents the sensitive location, which will be randomly displaced by a minimum distance of r1 and a maximum distance of r2. This results in the green point, which represents the masked location.

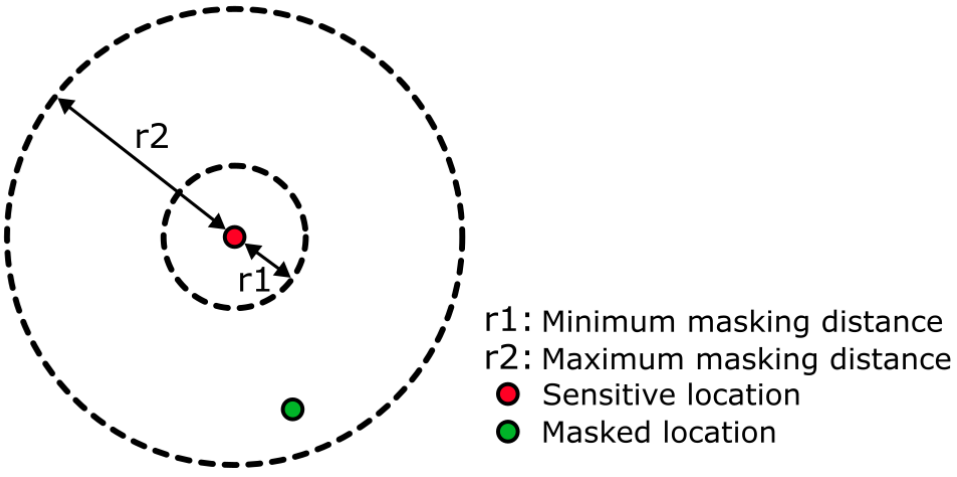


Figure 5: Donut Geomasking. Adapted from Hampton et al. (2010)

Location swapping is another geomasking method, where the original location is swapped with a new location which has similar geographic characteristics within the specified neighbourhood (Zhang et al., 2017). The researchers developed two techniques for location swapping. One technique is similar to random perturbation, while the other, location-swapping-with-donut, is similar to the donut geomasking method. The first method is based on drawing a buffer with a set radius around the original point. Then, this point gets randomly moved to another location in the generated buffer. What makes this method more unique than random perturbation is that the original point gets masked to an existing address location within the buffer area and the radius of the buffer gets adapted to the local population density (Zhang et al., 2017).

This privacy protection technique is illustrated in Figure 6. Initially, the original location is displayed. Subsequently, a buffer area is generated around the original location point. In Figure 6c, all the potential residential households that fall within the buffer area and can be swapped with the original point are identified. The original point is then removed from the dataset, and the swapped location is randomly selected from all the available locations within the buffer area.

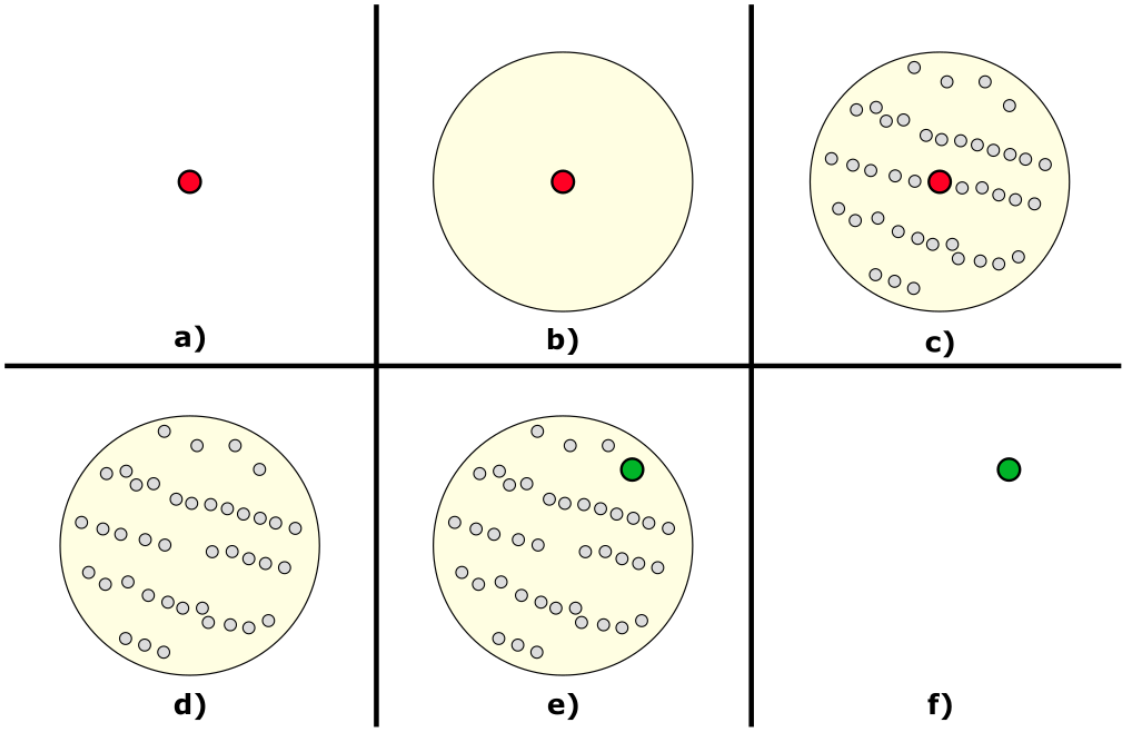


Figure 6: Location Swapping. Adapted from Zhang et al. (2017)

The second method developed, location-swapping-with-donut, is based on the same procedure as location swapping, except that a minimum masking distance like in the donut geomasking technique is utilised. The inner buffer's radius is a proportion of the external buffer's radius, and the buffer sizes vary based on the local population density in the residential area. One

characteristic of these techniques is, that if a buffer area includes a body of water or other unpopulated areas, the swapped location will not be placed in these areas, as there are no residential addresses available. As mentioned before, the masked location when using location swapping gets displaced to a geographically similar location into the buffer area, because address data is being used (Zhang et al., 2017).

This geomasking method is illustrated in Figure 7. As mentioned before, this method is very similar to location swapping, except that all the possible residential addresses are found within the donut buffer area. Also here, the swapped location is randomly selected from all the available locations.

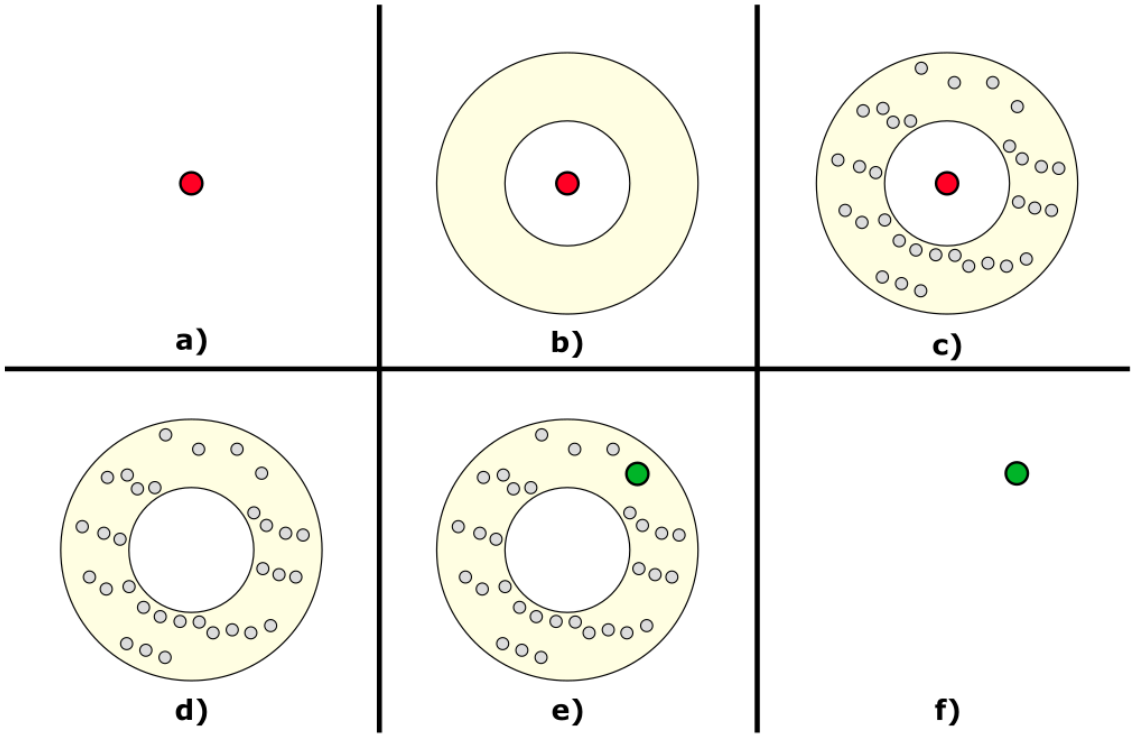


Figure 7: Location swapping with donut. Adapted from Zhang et al. (2017)

Richter (2018) also developed a geomasking method, the verified neighbour mask. It also utilises address data, similar to location swapping. However, instead of creating a buffer around the sensitive point, this technique selects for example the closest 30 addresses, and randomly moves each original point to a neighbouring address. Like in donut masking the nearest neighbours may be omitted from the dataset to avoid displacing the masked point too close to the original location.

Another geomasking technique that has been widely acknowledged is adaptive areal elimination (AAE). It does not utilise address data, like in location swapping or the verified neighbour masking method, to displace points, but uses census data and therefore, a better

privacy protection can be guaranteed. This technique works as follows: the geographic areas are being merged together until a defined minimum population has been achieved in the new area. If this has been achieved, every point in the area is being randomly moved within the new area (Kounadi & Leitner, 2016). As a result, the points are not being masked by a minimum or maximum distance, but randomly within each area (Swanlund et al., 2020a). An illustration of this geomasking method can be seen in Figure 8, wherein the orange polygons are aggregated until the population inside each area reaches 100 individuals per polygon.

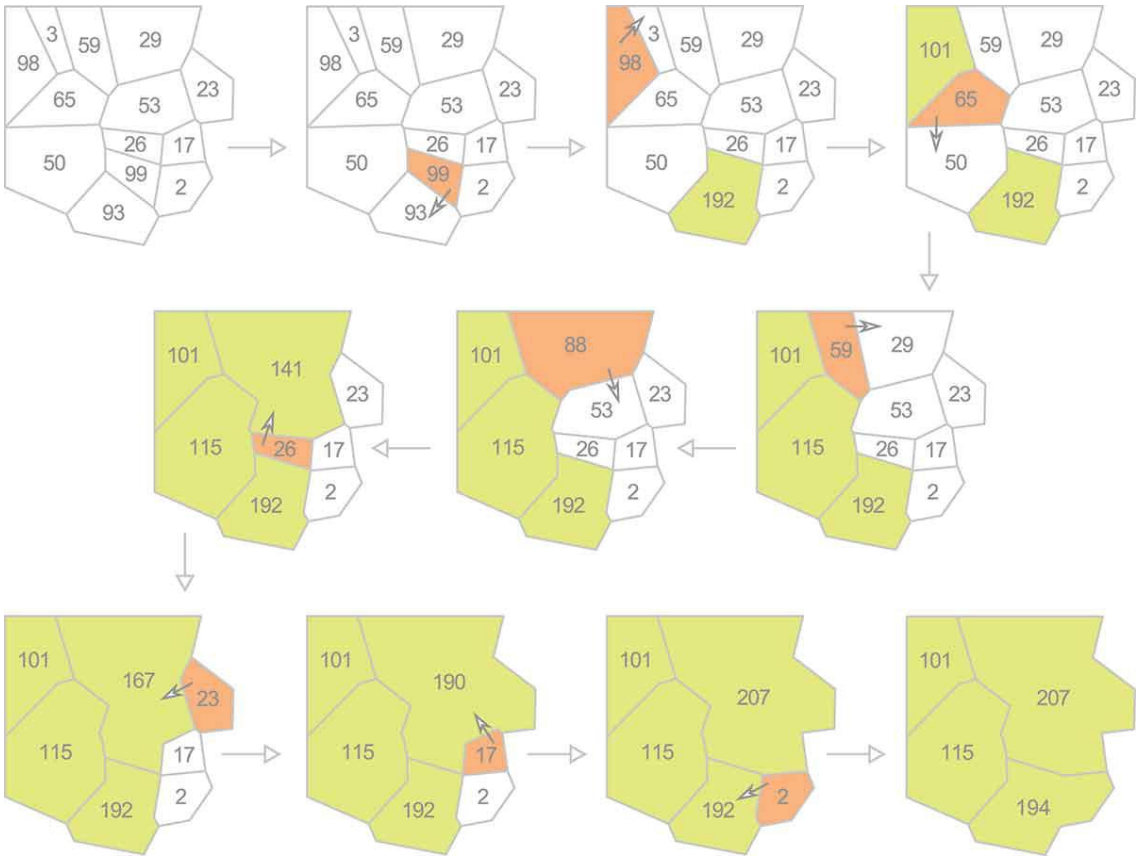


Figure 8: Adaptive areal elimination (AAE). Source: Charleux and Schofield (2020)

There are still many more geomasking methods that exist in the literature on location privacy. These include adaptive areal masking (Charleux & Schofield, 2020), which is based on the AAE approach from Kounadi and Leitner (2016), but adds an extra step that displaces the points in the aggregated areas. Street masking is a further LPPM that utilises road networks to anonymize location data (Swanlund et al., 2020b). Moreover, Seidl et al. (2015) introduced Voronoi masking (VM), a masking technique whereby each point is snapped to the nearest edge of the corresponding Voronoi polygon. Lastly, Polzin and Kounadi (2021) developed adaptive Voronoi masking (AVM). This privacy protection technique utilises the aggregation of polygons in AAE and the displacement method of the original location from VM.

All the above geographical masking methods were originally designed to protect location data, but some of them can also be a good choice for timed location data, such as fitness data. Because in these kinds of datasets the goal is to hide the confidential location data, like home and work address, from the users. So here, it is not the case that the goal is to prevent attackers from knowing the current or past location at a certain time (t), like it would be the case for masking mobile phone data.

In conclusion, multiple geomasking methods have been explored and demonstrated in this subchapter. Kounadi and Leitner (2014) found that a significant number of publications did not use masking methods to possibly protect personal sensitive information of individuals in the dataset. This next section will delve further into the privacy protection techniques provided by fitness tracking applications.

2.2.3 Privacy Protection Methods in Mobile Fitness Applications

As briefly mentioned in the first chapter of this thesis, mobile fitness tracking applications offer their users to track their sporting activities, such as running, hiking and cycling. Then, they can post their results on these applications, where athletes can compete against one another, but these recorded activities can also be shared on other social media networks. One example of a successful application is Strava. As of January 2022, Strava had approximately 95 million registered users and generated $220 million in revenue in 2022 (Curry, 2024). These applications can collect a huge amount of sensitive data from their users, such as health data, location information, temporal data and even valuable equipment that is being used. Therefore, it is important that these applications offer privacy protection mechanisms to protect these highly sensitive data (Hassan et al., 2018).

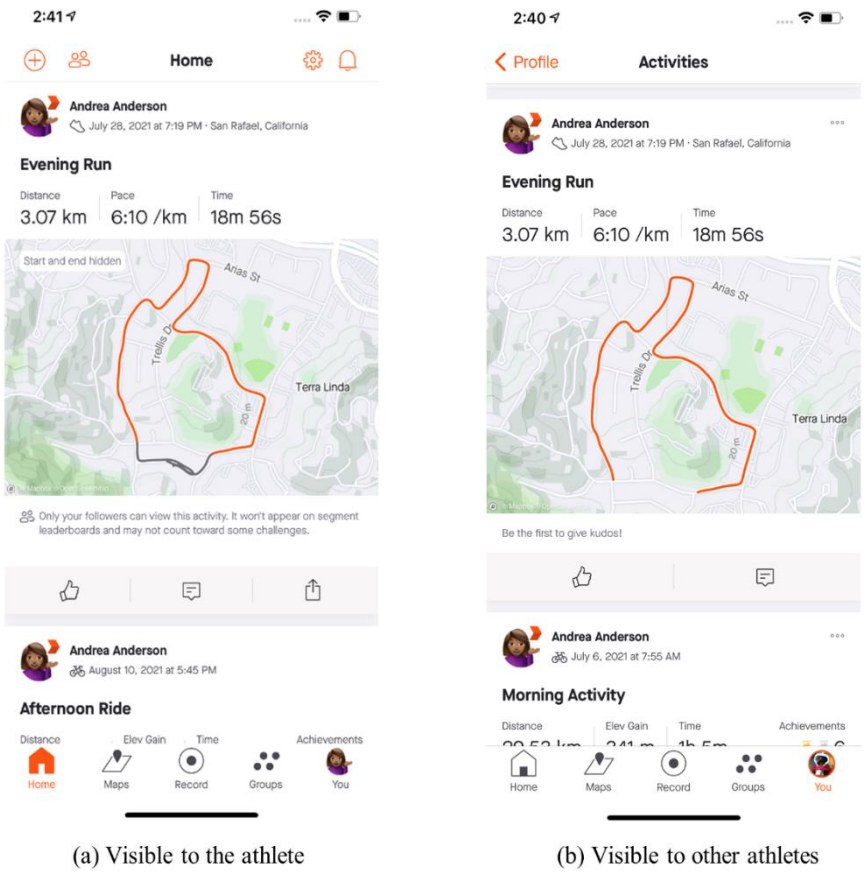
Because of these risks, some of the privacy protection methods available in mobile fitness applications are (Dhondt et al., 2022; Hassan et al., 2018):

1. **Private profiles and activities**[1] [2] [3] [4]**:** This option is very common among social media networks, where users can decide whether their profile and activities are visible to others or kept private. Subsequently, some applications allow the users to choose which activities should be private and which ones can be shared with their friends. However, using this option limits the functionality of the application, because private activities do not contribute towards global challenges and local leaderboards. This motivates users to share their activities publicly.

[1] https://support.strava.com/hc/en-us/articles/115000164850-Profile-Page-Privacy-Controls
[2] https://support.strava.com/hc/en-us/articles/216919377-Activity-Privacy-Controls
[3] https://help.runtastic.com/hc/en-us/articles/115003300785-Privacy-Settings
[4] https://support.garmin.com/en-US/?faq=VlB4wfnKqr2fChZJXF23a5

2. **Blocking users**[5] [6] [7]**:** Another common feature available to users on social media platforms, is the ability to block other users. The effect from blocking someone is different on every social media network, but it generally prevents any interaction between the blocked user and the person who got blocked.

3. **Endpoint privacy zone (EPZ)**[8] [9] [10]**:** Users often start and/or end their training session at a sensitive location and most applications give the users the option to hide the most sensitive parts of their route, which are within a certain distance of the sensitive location. If the start and/or end of the training route intersects with the defined EPZ, this intersected part of the training route is hidden from other users. Fitness tracking applications attempt to create a good balance between usability and an increase in privacy for the users with this method. This method is illustrated in figure 9.



(a) Visible to the athlete (b) Visible to other athletes

Figure 9: Visibility of a training activity for the user (a) and other users (b) on Strava where the user defined an EPZ. Adapted from Meg (2023)

[5] https://support.strava.com/hc/en-us/articles/216918327-Manage-Followers-and-Block-Athletes
[6] https://support.garmin.com/en-US/?faq=DGZI4GuXs80LHppO4wc6h8
[7] https://support.komoot.com/hc/en-us/articles/360047219052-Blocking-users
[8] https://support.strava.com/hc/en-us/articles/115000173384-Edit-Map-Visibility
[9] https://support.garmin.com/en-US/?faq=B9dlXYxQIr97DQwho5TBR7
[10] https://support.komoot.com/hc/en-us/articles/360046595312-Privacy-Zones

4. **Radius size of EPZs:** Most applications that have the EPZ technique integrated allow the users to define the radius of the EPZ. Some applications offer a minimum and maximum radius with fixed intervals.

As briefly mentioned in chapter 1, Hassan et al. (2018) tested how well this privacy technique protects a user's sensitive data and the risk of re-identification. As there are a handful of radii the circles can have, e.g. for Strava there are fixed intervals of 200m, and by every point that intersects by the EPZ, the higher the chance is that the user's sensitive location (centre of the circle) is re-identified. This is visualized in Figure 10, where in 10a two points/activities are placed along the border of the EPZ. The addition of a third activity results in the elimination of potential EPZs and the defined radius of the EPZ can be identified.

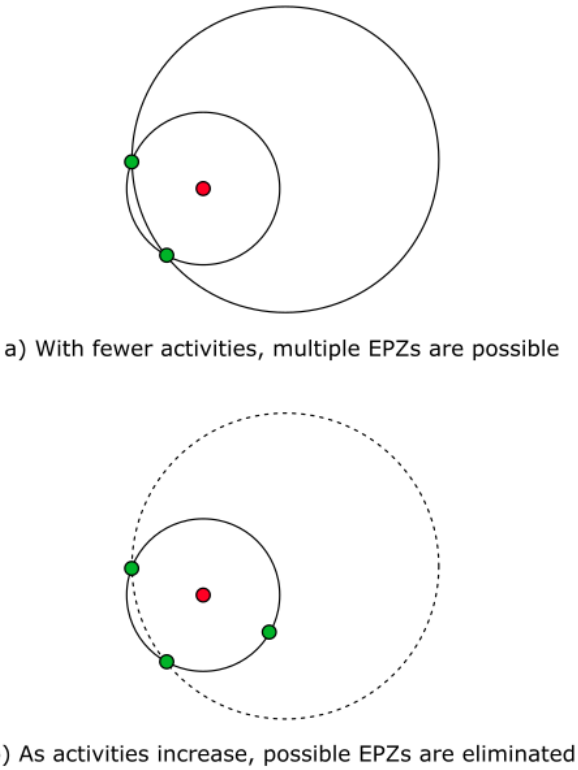


Figure 10: EPZ identification approach. The red point shows the original sensitive location, and the green points visualize the protected points. Adapted from Hassan et al. (2018)

The EPZs were searched and identified with a fitting circle algorithm. The dataset that the researchers web scraped from Strava had around 2,3 million EPZ-enabled activities from 432.022 users. The algorithm was able to identify 84% of the protected locations from users who had more than one EPZ-enabled activity. The accuracy of the inference attack increased to 95,1% for athletes with at least three EPZ-enabled activities. However, accuracy decreases as the radius of the zone increases. This is due to the fact that some training activities are wholly covered by the EPZ, which consequently reduces the usability for the user. This may be a contributing factor to the disfavouring of the larger radii. Only 44% of users who used a 1 km radius could be re-identified. It is worth noting, that the smaller radii are more popular than the

bigger ones. The researchers proposed three countermeasures to improve the anonymization of sensitive data and can be seen in figure 11 (Hassan et al., 2018).

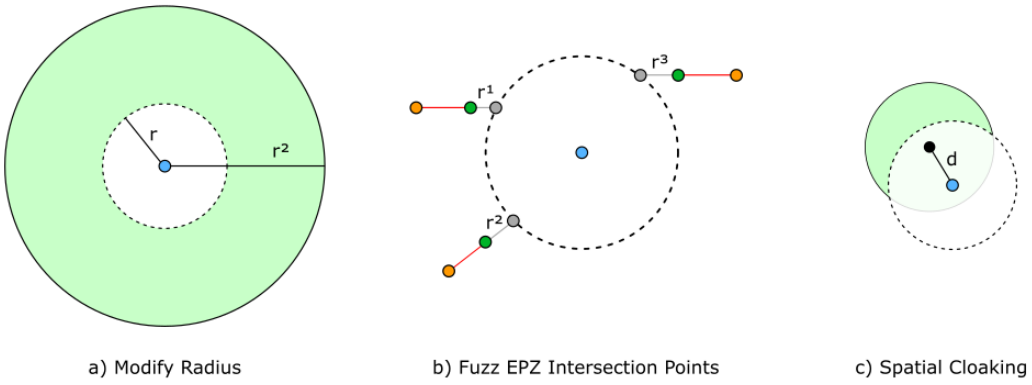


Figure 11: Three countermeasures for EPZs. Adapted from Hassan et al. (2018)

The first adjustment introduced by Hassan et al. (2018), is modifying the radius of the EPZs by making it bigger that can be seen in Figure 11a. Because of this change predicting the EPZ is harder, as the intersected points are further away from the sensitive location. As mentioned before, the accuracy of the attack decreased with a bigger radius, however, some routes are almost completely inside the buffer area and the tracked activity is basically invisible, which makes using a bigger radius unattractive for athletes.

The second modification involves adding some random noise to the data points which intersect with the EPZ, which is illustrated in Figure 11b. This can be achieved by removing some GPS coordinates which will move the points by a few meters in a random direction. The algorithm was adapted to this countermeasure and showed promising results and decreased the re-identification. Nevertheless, attackers could use a different approach to overcome the random fuzz used in the masking method (Hassan et al., 2018).

Lastly, a geomasking method called spatial cloaking could be introduced by fitness tracking applications, which is visualized in Figure 11c. Here, the centre of the circle gets moved by a certain distance and the sensitive location would be found inside the EPZ and not in the centre of the circle. The distance with which the EPZ gets shifted must be less than the radius of the EPZ, because otherwise the masked location could be found outside the protected zone. Using this countermeasure, a quick analysis also showed better protection, as it increases the complexity of re-identifying an EPZ (Hassan et al., 2018).

Hassan et al. (2018) used their algorithm on a small dataset collected from two other fitness tracking applications which also provide the EPZ privacy protection method and got the same results that they could successfully identify the sensitive locations of the users. Furthermore, they shared their findings with the involved fitness tracking applications, and they

acknowledged the vulnerabilities that were shown in their research. For example, Strava has limited the amount of user data that can be downloaded through their API and implemented the spatial cloaking method for each new EPZ created. Additionally, if a user is dissatisfied with their EPZ, they can re-randomize it.

Dhondt et al. (2022) conducted a review and evaluation of the adaptations made by some fitness tracking applications to the EPZ method, as recommended by Hassan et al. (2018). The researchers have confirmed that they made changes to the masking technique to better protect the sensitive location information of athletes who share their training routes on the social media aspect of the application. Nevertheless, some fitness tracking applications still do not provide users with the option to exclude the start- and endpoints within a defined area from their tracked training sessions.

Application	Downloads*	EPZ	EPZ Radius (meter)
Adidas Runtastic[11]	50M+	No	
Strava[12]	50M+	Circular	200 – 1600 (interval of 200m)
Garmin Connect[13]	10M+	Circular / Polygon	500 – 1500 (interval of 100m)
Komoot[14]	10M+	Polygon	Randomized**
Map My Run[15]	10M+	No	
ASICS Runkeeper[16]	10M+	No	
Nike Run Club[17]	10M+	No	
Relive[18]	10M+	Circular	200 – 1000 (interval of 200m)
Ride with GPS[19]	1M+	Circular	250, 500, 1000, 2000***
Map My Tracks[20]	100k+	Circular	500, 1000, 1500

** The data was extracted from the Google Play Store in July 2024.*

*** The shape and radius of the generated polygon is randomized and if the user is dissatisfied, a new EPZ can be generated.*

**** A small, random deviation is applied to the chosen centre of the EPZ.*

Table 3: Overview of popular fitness tracking applications with the number of downloads on the Google Play Store and the EPZ features.

A remaining problem with the EPZ method is that although users have applied an EPZ to their routes, the metric results of the tracked activities still show the full distance travelled to the other athletes in the training summary. Furthermore, if the athlete goes through the EPZ during

[11] https://www.runtastic.com/
[12] https://www.strava.com/
[13] https://connect.garmin.com/
[14] https://www.komoot.com/
[15] https://www.mapmyrun.com/
[16] https://runkeeper.com/cms/
[17] https://www.nike.com/gb/nrc-app
[18] https://www.relive.cc/
[19] https://ridewithgps.com/
[20] https://www.mapmytracks.com/

a training session, this part does not get deleted from the map visible to other users. Based on this knowledge Dhondt et al. (2022) wrote an algorithm to infer the sensitive location information of athletes. The researchers did this in two steps. First, they identify the EPZ and then they use a regression analysis to locate the protected locations of the users. The data collection took three months, as they had to download data with multiple users, because Strava limited how many datasets can be downloaded per day from their API. In the end, they collected around 1,4 million tracks from 400.000 athletes. Their algorithm is based on two inputs. Firstly, the road network is used to restrict the possible paths within an EPZ. Secondly, the length of the hidden paths within the EPZ can be calculated from the metadata. The inference attack is then based on a regression analysis, which estimates the protected location as a point at which the theoretical paths based on the reported distances best match (Dhondt et al., 2022). A visualization of the method can be seen in figure 12.

Figure 12: The illustration shows the distances travelled inside the EPZ with dashed lines, and the possible protected location is indicated by the intersection of the dashed lines at the black marker. Source: Dhondt et al. (2022)

As in Hassan et al. (2018), 84% of the EPZs with the smallest radius of 200 meters were efficiently identified by their approach. This radius is still the most popular among users. However, selecting a higher EPZ radius result in a decrease in the successful re-identification of protected locations by athletes. On Strava, users are limited to selecting a maximum EPZ radius of 1600 meters. The algorithm has the ability to re-identify 39% of sensitive location information when the largest radius is used (Dhondt et al., 2022). With these results, the researchers have managed to bypass the countermeasures that were recommended by Hassan et al. (2018) and implemented by some fitness tracking applications.

Dhondt et al. (2022) proposed six countermeasures to further enhance the anonymity of sites protected by EPZs. However, these measures come with a usability penalty for the users. The authors proposed four countermeasures to improve privacy protection in relation to the total distance travelled during an activity, which is the main focus of their inference attack. It was found that a generalisation of the total distance travelled in the training summary would be an effective countermeasure. This is because an attacker would not have detailed information about the actual distance travelled and therefore would not be able to determine the distance from the actual point and the masked start and endpoints. On the other hand, a random noise instead of rounding it up/down could be added to the travelled distance. This would create more adversity into the dataset, although, the authors argue that as soon as there are multiple activities from the same entry point into the EPZ the random noise could be averaged out. Another countermeasure is similar to one of the countermeasures of Hassan et al. (2018) to move the masked points either by a fixed or random distance but they propose to keep the total travelled distance the same. This would also add some randomness to the dataset, however, after a few training activities have been tracked this countermeasure may also be averaged out. Finally, the authors suggest truncating every single part of a training session (from the shared map and the total distance travelled) that is inside an EPZ. As previously mentioned, if an athlete returns to a defined EPZ during a training session, the track is not deleted as it is not a start- or endpoint. Also, these parts of a training session would be deleted.

Nevertheless, implementing any of these four countermeasures may have a significant impact on user usability. An important part of fitness tracking applications and their social network aspect are the training activities, sharing achievements and actively taking part in the local leaderboards against other athletes. Altering the total distance of an activity can result in either an overestimation or underestimation of a personal achievement for athletes, making it unattractive for most users (Dhondt et al., 2022).

The researchers have also presented two additional countermeasures that are more focused on EPZs. As seen by their results the effectiveness of an attack on a circular EPZ significantly decreases with a bigger radius. The smaller radii, such as 200 and 400 meters, are the most popular used ones among users. If a user uses a bigger radius (e.g. the maximum radius of 1600 meters on Strava) smaller training activities may be completely covered by this radius, which also makes it unattractive for athletes and negatively impacts the usability of the fitness tracking application. Furthermore, the last countermeasure discussed is allowing different EPZ shapes and making the zone around the sensitive location more complex. However, in the case of Dhondt et al. (2022), this would only make it more difficult to identify the EPZ. Nevertheless, it does not contradict the fact that the attackers still have knowledge of the total distance and therefore know the distance travelled within the EPZ.

To summarize their findings, making changes to the distance shown in the summary for other users, such as rounding the distance up or down, may be very effective but negatively effects

the usability of the users. On the other hand, making the EPZ more complex would somewhat protect users better and would not negatively affect athletes in using the application. The researchers shared their findings and suggested their improvements to the fitness tracking applications (Dhondt et al., 2022).

Furthermore, Mink et al. (2022) carried out a survey of EPZs. Participants received an explanation and training on the implementation of the geomasking method in fitness tracking applications. Approximately 75% of the participants currently use or have used a fitness tracking application. The majority of those who use or have used a fitness application share their training activities with other users on the application. However, around half of the participants expressed discomfort with sharing a map of their exercises, while the other half reported being somewhat or very comfortable. Additionally, only half of the participants used a privacy protection method provided by these applications.

After a brief explanation of what an EPZ is and how it generally functions, the participants were shown some training activities. They then performed an interference attack to locate an EPZ within the activity. When participants were provided with a map containing three activities and the most popular EPZ radius of 200 meters was used, they were able to accurately identify the masked location within a 50 meter radius. It was observed that participants who were provided with a map containing three activities expressed greater confidence in their estimation compared to those who were only presented with one activity on the map (Mink et al., 2022).

The participants were afterwards also asked about their opinions and impressions of an EPZ and the use of other privacy protection measures. They considered EPZs as an effective privacy protection method, but they also expressed that they would use EPZ in combination with other privacy precautions such as setting their profile to private or setting singular posts to private (Mink et al., 2022).

The current state of research about EPZs has been reviewed in this subchapter. The following subsection explores and explains the privacy protection metrics used to evaluate the risk of re-identification, namely k-anonymity and spatial k-anonymity.

2.2.4 Re-identification and (Spatial) k-anonymity

When data owners anonymize a dataset to protect sensitive data, there is always a risk of an attacker identifying the natural person whose information was obfuscated. The k-anonymity metric is used to evaluate the level of privacy protection provided by an anonymized dataset. This metric was first introduced by Samarati and Sweeney (1998) for tabular data. K-anonymity is a technique where at least a particular number of individuals (a value of k) have to share a similar set of attributes (e.g. date of birth, gender, sex etc.). As a result a specific individual cannot be uniquely identified in the dataset (McKenzie et al., 2022; Zhang et al., 2017).

Tables 4 and 5 provide an example of how the k-anonymity technique works and how it can be used to anonymise data while preserving the ability to analyse it further. As previously mentioned, quasi-identifiers such as name, gender, date of birth, and postcode can be used to identify an individual. To protect a specific individual from being identified these quasi-identifiers are being obfuscated in the dataset. However, researchers can still analyse the sensitive attribute of disease in the dataset.

SSN	Name	Gender	Birthdate	Postcode	Disease
1111010680	Max	Male	01.06.1997	1010	Liver
3333050901	Louisa	Female	05.06.1998	1010	No Illness
2222200295	Patrick	Male	20.06.1995	1010	HIV/AIDS
4444280289	Lisa	Female	20.02.2001	2010	Cancer
5555251273	Valerie	Female	25.02.2005	2010	No Illness
6666560987	Franz	Male	01.02.2002	2010	Liver

Table 4: Non-Anonymous health dataset

Table 5 shows that a k-anonymity of 3 (k=3) was achieved by removing the social security number, name and gender from the dataset and generalizing two quasi-identifiers (date of birth and postcode).

SSN	Name	Gender	Birthdate	Postcode	Disease
			.06.19	10**	Liver
			.06.19	10**	No Illness
			.06.19	10**	HIV/AIDS
			.02.20	20**	Cancer
			.02.20	20**	No Illness
			.02.20	20**	Liver

Table 5: Anonymized health dataset (k=3)

The research of Sweeney (2000) revealed that a person's gender, date of birth, and postcode alone could identify 87% of the US population. Furthermore, approximately half of the population of the United States can be identified by their gender, date of birth and place of residence (city or town).

However, to assess the effectiveness of geographic masking methods in protecting sensitive data, spatial k-anonymity was introduced. This technique is an extension of k-anonymity (Zhang et al., 2017). Spatial k-anonymity measures the amount of individuals, households, or addresses who are located in the same geographical area as the masking method to evaluate the probability of re-identification (Wang & Kwan, 2020). Depending on the masking method used, some of which have been explored in section 2.2.2, the masked location could theoretically be moved near to the original location. Nevertheless, when it comes to spatial k-anonymity, the

number of other individuals in the surrounding area is more important than the distance displacement. Location privacy protection mechanisms should provide a high level of spatial k-anonymity (Zhang et al., 2017).

A great example is that in urban areas a high level of spatial k-anonymity can be achieved as the population density is high. On the other hand, achieving a high level of spatial k-anonymity in suburban areas generally requires a greater distance displacement due to the lower population density.

An example of how (spatial) k-anonymity is calculated, is visualized in Figure 13. The red dot is the original location, and the green dot is the masked location. Next, a buffer is drawn around the masked location and the grey dots in the buffer area are the other possible locations that could have been used as the masked point. Finally, the total number of other possible location to which the point could have been displaced are summed up. The result is the spatial k.

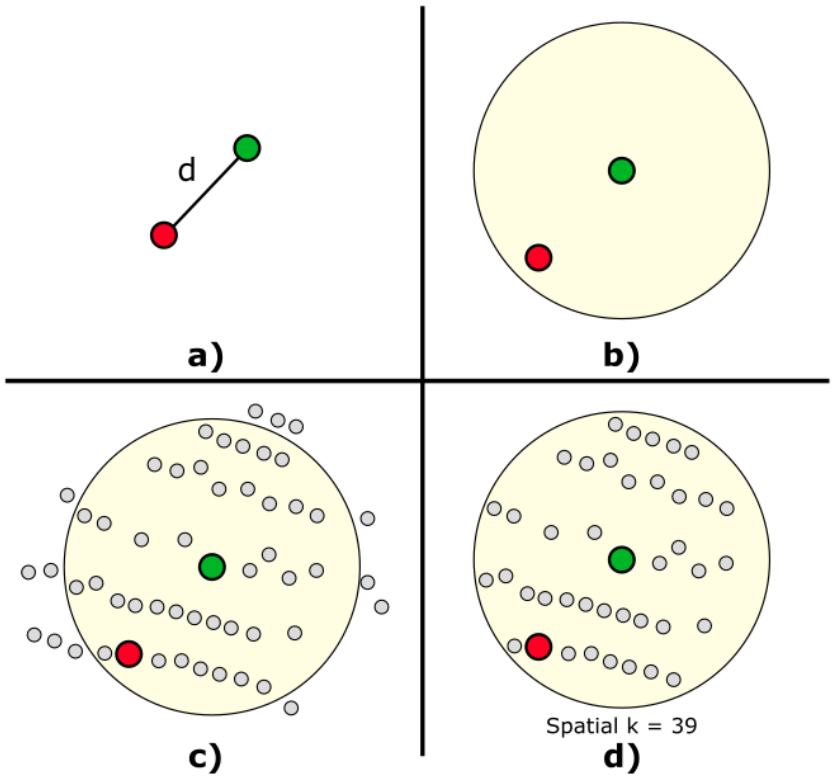


Figure 13: An illustration of calculating (spatial) k-anonymity. Adapted from Zhang et al. (2017)

The privacy protection method EPZ, which is employed in certain mobile fitness tracking applications, does not take into account the concept of spatial k-anonymity. In areas of low population density, the risk of re-identification remains high, given that there may only be a single building or address within the defined EPZ. It is therefore imperative to consider spatial k-anonymity as an additional factor.

3 Methodology

After the analysis and introduction of several geomasking methods, this chapter presents the methodological framework of this thesis. Firstly, the software used to create, analyse and present the privacy protection approach are described. Secondly, the areas of interest selected for testing the ska-based method are presented, accompanied by some geographical information about them. Moreover, the data that was simulated and utilised is described in the subsequent sections. Finally, the ska-based privacy protection method is demonstrated and explained.

A general workflow of the methodology can be seen in figure 14.

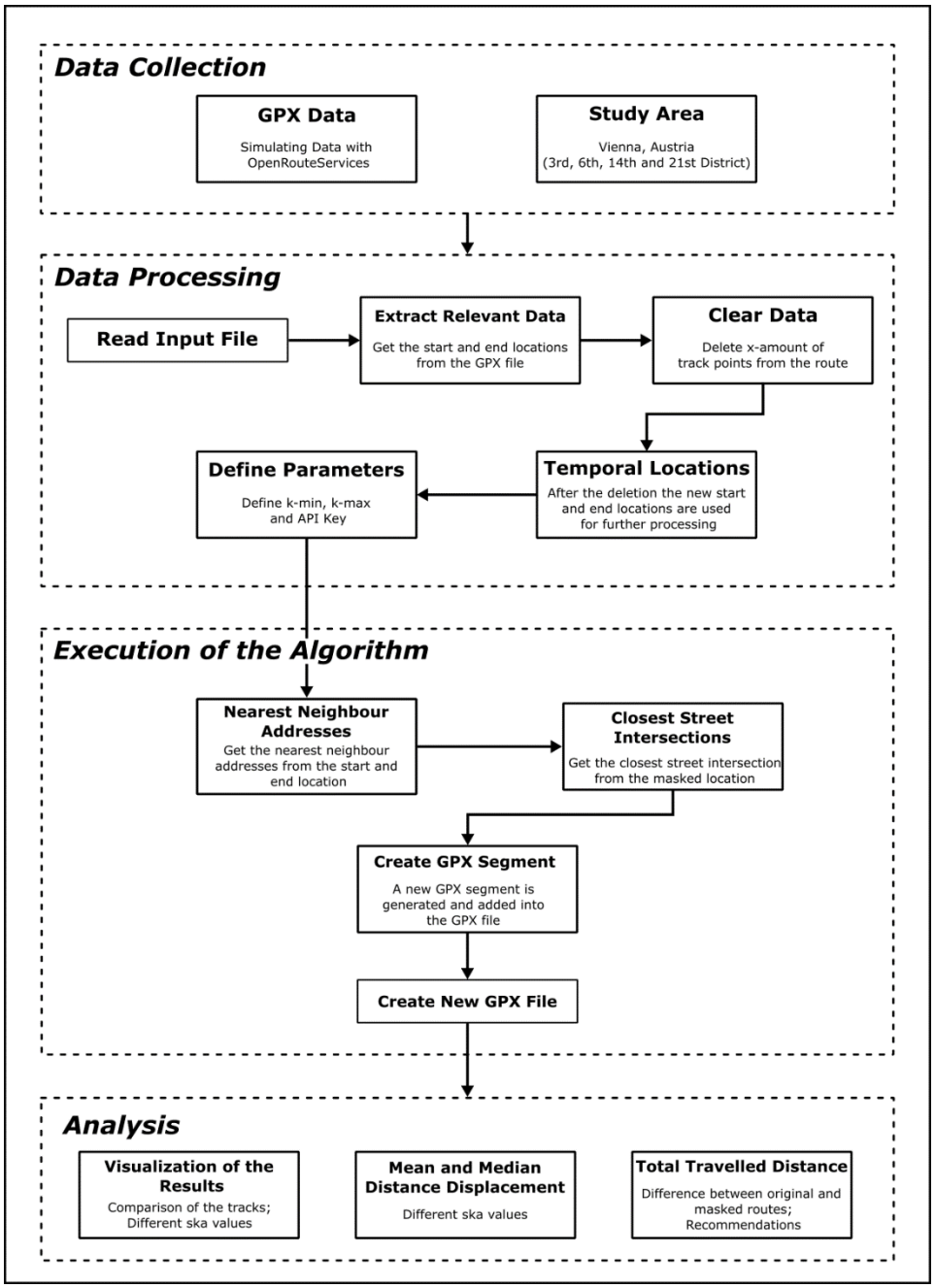


Figure 14: Flowchart of the practical part of the thesis

3.1 Software

This section provides an overview of the various software that were used in the empirical part of this thesis. The Python programming language was employed throughout this thesis for the development of a ska-based method for the protection of sensitive location data associated with tracked routes in a fitness tracking application. Furthermore, the Openrouteservice API is used for the simulation of running tracks within the specified areas of interest. Subsequently, this same API is utilised in the privacy protection algorithm to generate segments for the adapted track. The visualization of the original and masked running tracks and the creation of maps for this thesis were conducted using ArcGIS Pro. Lastly, a repository has been created on GitHub for the purpose of viewing the ska-based privacy protection mechanism and replicating the workflow described in this thesis.

3.1.1 Python

Python[21] is an open-source programming language which was used to develop the ska-based privacy protection method. Various python libraries, such as gpxpy, geopy and Shapely were used to read, manipulate and process the geographical data.

3.1.2 Openrouteservice

The Openrouteservice API[22] is an open-source software application that utilises crowdsourced geographical data from **OpenStreetMap** (OSM)[23], which is generated and collected by the general public and OSM contributors. This API offers different services, such as calculating Isochrones, Time-Distance Matrices, adding elevation information to point or line data etc. (Openrouteservice, 2024). The directions service was used for the purpose of data acquisition, with the objective of simulating data of running routes. It is also integrated into the privacy protection approach, thereby enabling the generation of routes between two new points, the original location and the masked location.

3.1.3 ArcGIS Pro

ArcGIS Pro[24] is a powerful desktop GIS application by Esri that provides a wide range of tools for data creation, analysis, exploration, and visualization (Esri, 2024a). ArcGIS Pro was used in this thesis to visualize the running tracks and for the creation of maps. This software was used with a license that is provided by the university of Vienna.

[21] https://www.python.org
[22] https://openrouteservice.org
[23] https://www.openstreetmap.org
[24] https://www.esri.com/en-us/arcgis/products/arcgis-pro/overview

3.1.4 GitHub

GitHub[25] is a cloud-based open-source platform that allows developers to share, work and store their code on (GitHub, 2024). The python script of the ska-based privacy protection method developed in this thesis is available in a repository[26] on GitHub, so that other users can look and review the code. Additionally, this code can be utilised and improved in further studies.

3.2 Study Area

The study area chosen in this master's thesis is the city of Vienna in Austria. Vienna is the country's capital city and is situated in the eastern region of the country. With a population of approximately two million, it is the largest city in Austria. The total area of Vienna is around 420 km² and this results to a population density of around 5800 inhabitants per km². Furthermore, Vienna consists of 23 districts and approximately 45% of the area is classified as greenspaces (City of Vienna, 2024).

The population density varies across Vienna, which is illustrated in Figure 15 and Table 6. The city's inner districts typically display a higher population density, while the outer districts often have a lower population density. Not only do the districts differ in terms of population density, but also in terms of geographical aspects, such as parks, forest and river, and the relationship between private households and businesses. In order to examine the performance of the ska-based privacy measure, four districts with different characteristics were selected.

First of all, two districts with a high population density and urban features were chosen. The third district, Landstraße, was selected, because it is an inner district and therefore close to the city centre. Moreover, this district has several greenspaces, such as Stadtpark and the Belvedere, as well as a riverside location adjacent to the Danube River. Furthermore, Landstraße had a total population of 93.756 inhabitants in 2023, distributed across an area of 7,4 km². This results in a population density of approximately 13.100 individuals per square kilometre (City of Vienna, 2024).

Additionally, the sixth district, Mariahilf, was chosen to investigate the ska-based privacy approach. This municipality is located next to the first district and includes the biggest shopping street in Vienna, the Mariahilfer Strasse. In comparison to the third district, Mariahilf has little greenspaces and is mostly made up of buildings with many businesses scattered around the area. Regarding the population, the sixth district had 31.423 inhabitants in 2023, spread across an area of 1,46 km². As a result, the population density is around 21.600 residents per square kilometre (City of Vienna, 2024).

[25] https://github.com/
[26] https://github.com/Digital-Geography/ska-based-LPPM

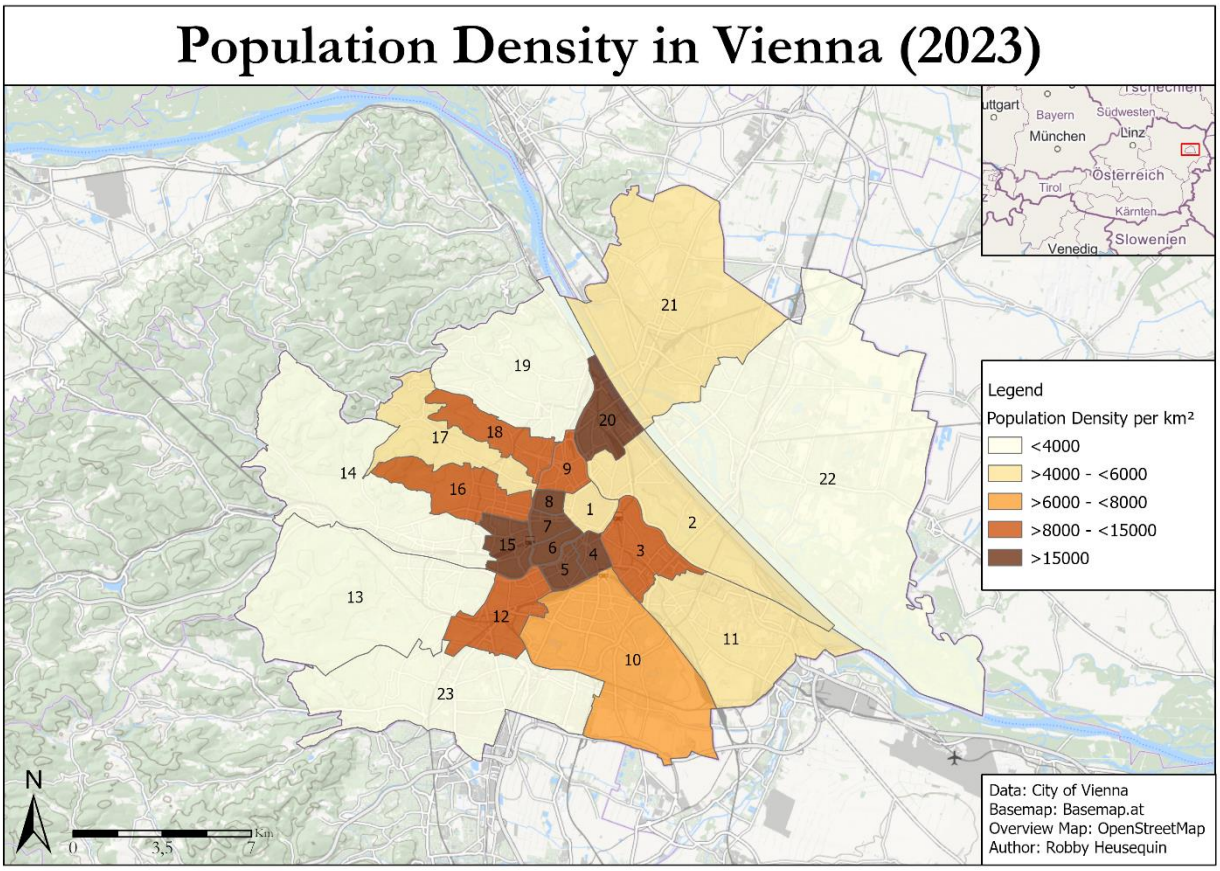


Figure 15: Map of the population density in Vienna (2023)

Subsequently, two outer districts with a lower population density and more suburban characteristics were chosen. One of these municipalities is the 14th, Penzing. It is located in the western part of Vienna and a large area is made up of greenspaces, such as a part of the Vienna woods. Penzing has a total area of 33,76 km² and had a total population of 96.828 in 2023. The population density is about 2900 inhabitants per square kilometre, which is the second lowest in Vienna (City of Vienna, 2024).

Lastly, the 21st district of Florisdorf, which is found in the northern part of Vienna, was selected as the final area of interest. As in the 14th district, Florisdorf consists of multiple relaxation areas. Furthermore, this municipality borders the Danube, and a section of the New Danube and the Danube Island are located within its boundaries. Florisdorf has the third highest total population of Vienna with 183.895 residents in an area of 44,44 km². This results to a population density of about 4100 individuals per square kilometre (City of Vienna, 2024).

District Code	District Name	Total Population	Area (km²)	Population Density (per km²)
1010	Innere Stadt	16.620	2,87	5793,42
1020	Leopoldstadt	108.269	19,24	5626,7
1030	Landstraße	96.756	7,4	13.079,13
1040	Wieden	33.633	1,78	18.945,95
1050	Margareten	55.018	2,01	27.349,84
1060	Mariahilf	31.423	1,46	21.595,92
1070	Neubau	31.581	1,61	19.636,9
1080	Josefstadt	24.674	1,09	22.637,83
1090	Alsergrund	42.206	2,97	14.223,3
1100	Favoriten	218.415	31,83	6862,23
1110	Simmering	109.038	23,26	4688,53
1120	Meidling	100.281	8,1	12.375,4
1130	Hietzing	55.568	37,71	1473,38
1140	Penzing	96.828	33,76	2867,85
1150	Rudolfsheim-Fünfhaus	76.109	3,92	19.423,18
1160	Ottakring	102.444	8,67	11.811,76
1170	Hernals	56.033	11,39	4918,97
1180	Währing	51.559	6,35	8123,11
1190	Döbling	75.517	24,94	3027,48
1200	Brigittenau	85.690	5,71	15.005,76
1210	Florisdorf	183.895	44,44	4137,75
1220	Donaustadt	212.658	102,3	2078,78
1230	Liesing	117.882	32,06	3676,72

Table 6: Population density of the districts in Vienna (2023). Data from City of Vienna (2024)

3.3 Data

In order to apply the ska-based privacy method, a GPS Exchange Format (GPX) file representing the tracked route with confidential or sensitive data is required. A GPX file is based on Extensible Markup Language (XML) and is capable of storing GPS data that can include a description of the location, time, altitude and the track, which consists of X and Y coordinates in the coordinate system World Geodetic System 1984 (WGS84) (Xie, 2021).

To examine the algorithm, it was decided to simulate fake data rather than use existing GPX data with sensitive location information, which makes it possible to share the results. Moreover, this does not limit the evaluation phase of the ska-based approach, as the data can be generated in a manner that closely resembles real data. As briefly mentioned above, the synthetic GPX data in the defined areas of interest in Vienna were simulated using the Openrouteservice API. In addition, as many different possibilities as possible were taken into account in the data simulation process, such as routes with a different start and end location, routes with the same start and end location, various lengths of the routes, diverse route patterns, the neighbourhood characteristics (urban / suburban) and geographical aspects (forest, parks, river etc.).

A total of 16 routes with different characteristics as described above, were created for this study. These routes were simulated in the four areas of interest in Vienna. Within each district, four routes were created. Among these, two routes that have a different start and end location and two routes have the same start and end location.

Route ID	District	Same or Different Start and End Location	Urban / Suburban	Length of Route	Geographical Aspects
1	Landstraße	Different	Urban	1,37 km	Park
2	Landstraße	Different	Urban	2,59 km	Parks, River
3	Landstraße	Same	Urban	4,09 km	River
4	Landstraße	Same	Urban	3,60 km	Parks
5	Mariahilf	Different	Urban	1,45 km	
6	Mariahilf	Different	Urban	2,24 km	Park
7	Mariahilf	Same	Urban	4,74 km	Parks
8	Mariahilf	Same	Urban	4,12 km	Parks
9	Penzing	Different	Suburban	1,22 km	
10	Penzing	Different	Suburban	2,65 km	Park
11	Penzing	Same	Suburban	4,80 km	Forest
12	Penzing	Same	Suburban	5,42 km	Forest
13	Florisdorf	Different	Suburban	3,84 km	Park
14	Florisdorf	Different	Suburban	2,78 km	Park, River
15	Florisdorf	Same	Suburban	4,46 km	
16	Florisdorf	Same	Suburban	5,82 km	Park, River

Table 7: Characteristics of the simulated routes in Vienna

To be able to properly use a GPX file for the algorithm, the encoding has to be without a byte order mark (BOM) and no waypoints can be included in the file, as these are not being processed. When a GPX file is encoded with BOM, the python library utilised to read the file will generate an error message indicating that the XML format is not well-formed. Removing the BOM encoding can be done with the python code that is attached in Appendix A or which can be found in the GitHub repository.

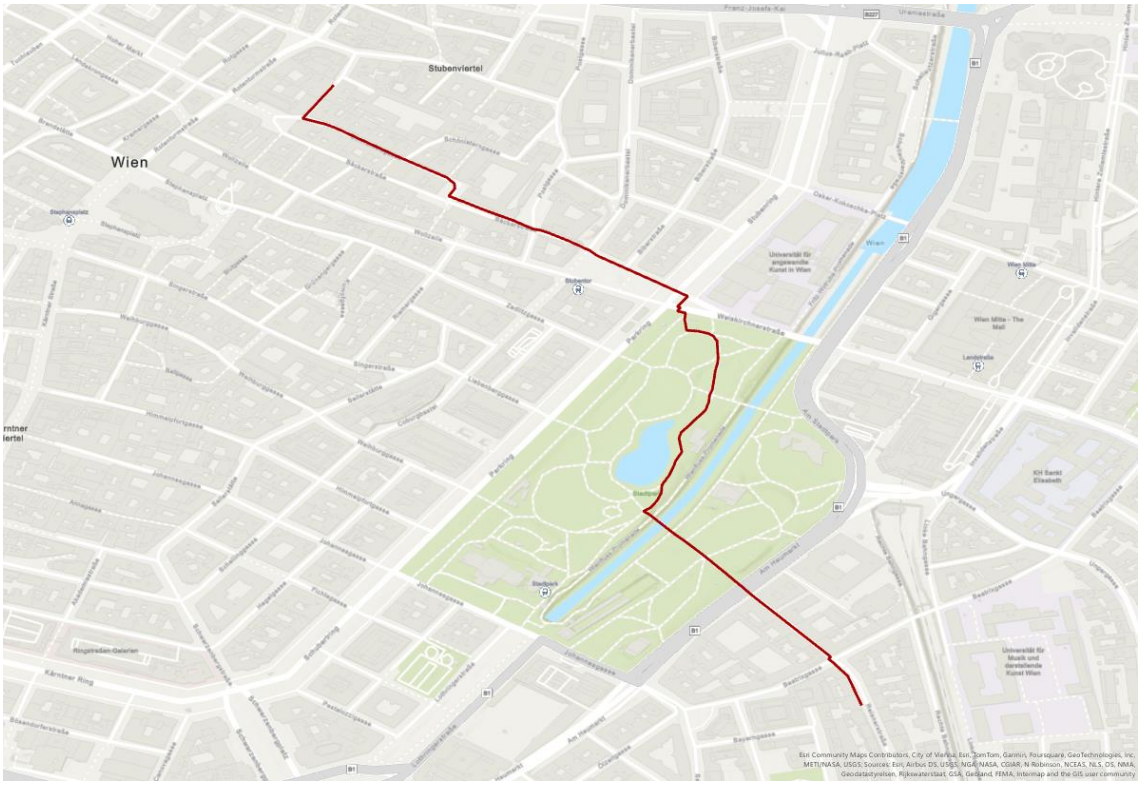


Figure 16: Example of a GPX track

Furthermore, to work with address data in the ska-based approach the **Overpass Turbo API**[27] is utilised. The Overpass API is a database engine that can be used to query OSM data in a read-only mode (OpenStreetMap, 2024). In addition to its use in querying the nearest addresses around the confidential start and end points, this API is also utilised in obtaining the closest street intersections.

As mentioned above the Openrouteservice API is also used in the algorithm to create two new segments in the route. Once from the original start location to the masked start location and also from the original end location to the masked end location. In order to successfully make requests with the API, it is necessary to create an account on the Openrouteservice website and generate a personal token, which can be done for free.

Finally, an open source geolocator called Nominatim, which is also based on OSM data, is used to obtain the X and Y coordinates of an address.

[27] https://overpass-turbo.eu/

3.4 Ska-based Privacy Protection Mechanism

This section presents the ska-based privacy protection method for fitness tracking applications, which displaces the sensitive location information by address data. The aim of this new geomasking method is to improve the anonymization of confidential location information and brings the following advantages: first, it allows users to define a level of spatial k-anonymity, thereby reducing the risk of a potential re-identification. Second, by using address data instead of displacing the sensitive locations by distance, it prevents the masked location from being relocated to an invalid location, such as a water body or in forests. Third, an extra step is added to the masking algorithm. After the masked address is selected, the point is moved to the nearest street intersection to reduce the risk of false re-identification, and this also adds an extra protection as the location now may correspond to more than 1 address (typically 2 to 4). Lastly, a new segment is added to the route, which prevents an attacker from guessing where the original start and end points are.

The level of spatial k-anonymity is defined by a k-min and k-max value. The k-min value is mainly used to add an initial error to the original sensitive location and should always be lower than the k-max value. Furthermore, if there were a single k-value and the attacker were aware of this value, it would be easier for the attacker to infer the sensitive location information from multiple masked copies of the same sensitive location. However, the spatial k-anonymity level can actually be higher than the user-defined level, because a segment is added to the route, which increases the spatial k-anonymity level.

As a consequence of the modifications made to the tracked route by the algorithm, the total distance travelled is prone to adjustments. As was examined by Dhondt et al. (2022), when the original distance travelled is retained and shared with other users, it is vulnerable to re-identification. It was therefore necessary to implement a solution that would be suitable for the user.

Modifying the total distance of a training activity can either result in an underestimation or an overestimation of the performance of the athlete, which generally makes the protection method unattractive for them to use it. In the algorithm, the total distance travelled is calculated for both the original and the masked routes. A possibility to handle the total distance travelled would be that, when the route is shared on the social network platform, the total distance from the masked, not the original, route is shared. Subsequently, when a certain number of activities are tracked and the athlete has not recorded the same trajectory each time, the original distance travelled from all these activities can be aggregated. This measure could counter an under- or overestimation and as a result, the athlete would still be eligible to participate in competitions hosted in the fitness tracking application and be included on the leaderboards.

The Python code is displayed in Appendix B or in the GitHub repository, linked above. Next, is a brief detailed description of the steps in the ska-based privacy method:

Step 1: **Input File and Extract Data**

The script takes a GPX file without BOM encoding with one trip as an input. The start and end locations are extracted from the GPX file, as they are potentially sensitive/private locations (e.g. residence or work location). Then, the first and last five track points are deleted from the original GPX track and the new temporary start and end locations are also exported for further processing.

Step 2: **Define Parameters**

The k-min and k-max values are defined by the end user. Furthermore, the API key for Openrouteservice must be defined as well.

Step 3: **Obtain the Nearest Neighbour Addresses**

The user-defined k-min and k-max values are used to obtain the nearest neighbour addresses. First, the k-min nearest addresses are saved and a random address from this list is selected. Then, the k-max nearest addresses from the chosen k-min nearest address are obtained. Again, a random address from this list is selected. This results in the masked location. In this process, it was possible for the original address to be in the result list, because the same address is in the OSM dataset with a slightly different set of coordinates. To counter this, all the coordinates of the addresses are compared to the original address with a small tolerance. Finally, the same is done for the other addresses. The k-nearest addresses are stored in a Python set that does not allow duplicate values.

Step 4: **Obtain the Closest Street Intersections**

The nearest street intersections from the masked address (k-max nearest address) are acquired and the closest one is selected. Finally, this street junction is used as the masked point. In order to be able to find the nearest street intersection, the bounding box has to be defined. This is done by extracting the minimum and maximum latitude and longitude from the route segment and adding a small, fixed buffer because the masked address could be outside the bounding box of the original route.

Step 5: **Creation of a New GPX Segment**

A new GPX segment is generated from the closest street intersection (of the k-max randomly selected nearest address) to the new temporary address (this is not the original address, as the first and last 5 points in the GPX file were removed).

Step 6: **Modification of the Input GPX File**

The new GPX segment is added at the correct position in the input GPX file. The masked start segment is added at the first position and the masked end segment is added at the third position in the GPX file.

Step 7: **Repeat**

First, steps 3 to 6 are executed for the starting location and are then repeated for the end location.

Step 8: **Save the New GPX File**

Finally, a new GPX file with the adjusted track is saved.

Step 9: **Calculate the Original and Masked Travel Distance**

The original and masked travelled distance of the GPX files is calculated, because these will be used for further use.

The primary difference between the EPZ geomasking approach commonly employed in fitness tracking applications and the developed ska-based privacy protection method in this study is that, in addition to the removal of a sensitive portion of the original track, a new segment is incorporated into the track, thereby further enhancing the protection of the user's privacy. The goal is to obscure the sensitive locations of the track more effectively and make it more challenging for the attacker to infer the original start and end locations of the athlete. Furthermore, the ska-based protection approach tries to enhance the usability of the masked trajectory and make it more attractive for individuals to use. Lastly, as shown before, the EPZ geomasking approach uses distance, whereas the ska-based privacy protection method employs addresses and street intersections.

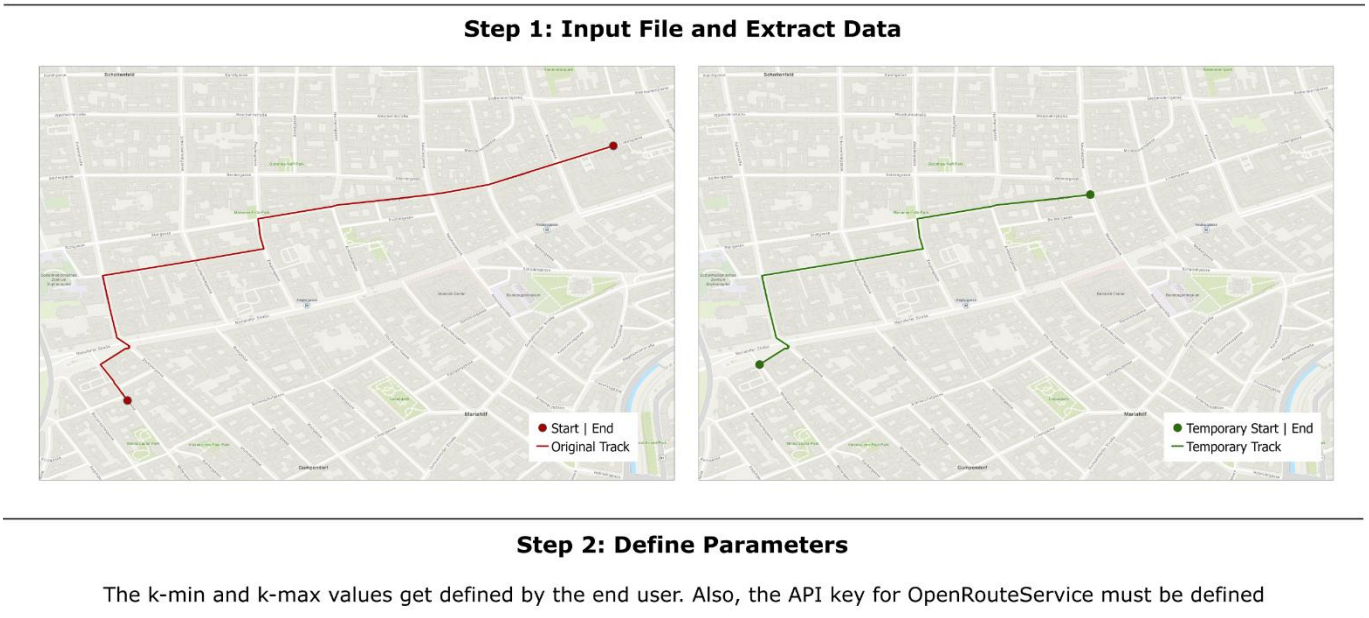


Figure 17: Visualization of the ska-based privacy protection method (steps 1-2)

Step 3: Obtain the Nearest Neighbour Addresses

Here k-min = 5 Here k-max = 20

Step 4: Obtain the Closest Street Intersection

The closest intersection (purple dot) is used as the masked location

Step 5: Creation of a New GPX Segment

Step 6: Modification of the Input GPX File

The new GPX segment is added to the temporary track

Figure 18: Visualization of the ska-based privacy protection method (steps 3-6)

Step 7: Repeat

Steps 3 to 6 are executed for the starting location and are repeated for the end location

Step 8: Save the Masked Track

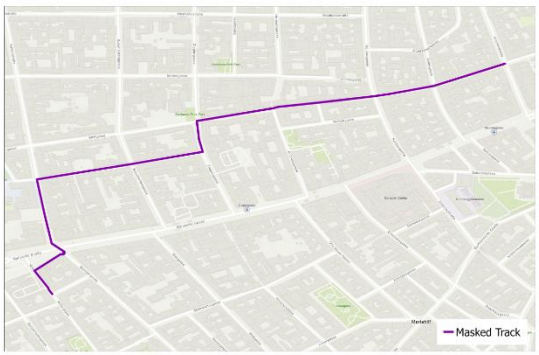


Figure 19: Visualization of the ska-based privacy protection method (steps 7-8)

3.5 Evaluation of The Geomasking Performance

This chapter briefly describes the methods used to test the performance of the ska-based privacy approach.

3.5.1 Visualize The Results of The Outcome

To explore the original track and compare it to the masked track, both are visualized and compared using ArcGIS Pro. For a GPX file to be visualized in ArcGIS Pro the tool “GPX To Features” is used, which converts the point data in the GPX file to features. Moreover, the tracks will be converted to polylines (Esri, 2024b).

3.5.2 Explore The Mean and Median Distance Displacement

In order to examine the mean and median distance displacement of the masked data points, the geodesic distance between the original location and the closest street intersection from the masked location is determined. This is achieved through the use of the geopy library in Python. The geodesic distance calculates the shortest path between two points based on an ellipsoidal model of the earth (Karney, 2013).

3.5.3 Different Spatial k-anonymity Values

As mentioned above, the ska value is user-defined. In this work two different k-min and k-max values are tested and evaluated. First, a k-min value of 5 and a k-max value of 20 are chosen. Second, the level of ska was increased with a k-min value of 10 and a k-max value of 50. This is done by changing the input values in the script.

4 Results and Discussion

This chapter presents the results of the ska-based privacy protection method and discusses the effectiveness of the outcome. First, the different behaviour patterns that emerged during the analysis of the results from the ska-based geomasking method are visualized (see section 4.1). Thereafter, the mean and median of the distance between the original locations and the masked locations are explored and examined (see section 4.2). The results from the two distinct levels of ska are integrated into the respective subchapters. Lastly, a suggestion on how to handle the original and masked travelled distance and time metrics is proposed. (see section 4.3).

It is important to note that the goal of the ska-based privacy protection method is to mask the route in a way that the original start and end locations, which are treated as sensitive locations, cannot be re-identified. Furthermore, the geomasking method adapts the track in a way that it can still be shared with other athletes.

4.1 Visualization of the Original and Masked Routes

To evaluate the efficiency of the ska-based geomasking method, it is essential to visualize and compare the original routes with the masked routes. A visual exploration of the original and masked routes allows for the identification of the specific differences between the two. Moreover, the results of the algorithm can be analysed and thereby evaluated if the geomasking method was successful or not. Additionally, the different types of modifications observed can be categorized. This makes it possible to better understand the strengths and weaknesses of the ska-based privacy protection method, which ultimately provides future improvements that can be made to make the geomasking method more efficient and user friendly.

This structure makes it possible to gain a deeper insight into and a better understanding of the ska-based geomasking technique. Furthermore, as mentioned above, the main difference between the four simulated routes in each district is that two have the same start and end point and the other two have a different start and end points. These are evaluated separately in the next subchapters.

It should be noted that the colours used in the following maps for the GPX tracks may vary and indicate the various iterations of the algorithmic process. The original route is always represented by red, the first iteration by green, the second by purple and the third by brown.

All the simulated routes used for testing the ska-based geomasking method can be found in the GitHub repository linked above.

4.1.1 Different Start and End Location

To begin, the results of the routes with a distinct start and end location are examined. In general, the developed ska-based geomasking method shows promising results for the protection of the sensitive location information and is effective in visually masking the start and end points, which provides a good usability for the athletes. This was the case for both levels of ska that were tested with these routes. Moreover, the outcomes show promising results in the four defined areas of interest in Vienna.

Firstly, the diverse behaviours observed during the evaluation phase of the geomasking method, particularly in scenarios where the method demonstrated optimal performance, are examined and presented. The figures 20-21 show the outcome of specific masked routes, where the original route was either shortened and/or extended. A reduction in the length on one or both sides of the masked route means that the original start and/or end location are no longer included in the masked route. On the other hand, when the route is extended, the original start and/or end location remains in the masked track. Nevertheless, the attacker is unable to identify, whether the track has been shortened or extended. In either case, this behaviour is effective for the protection of sensitive locations while making it more difficult to re-identify the individual from the masked data.

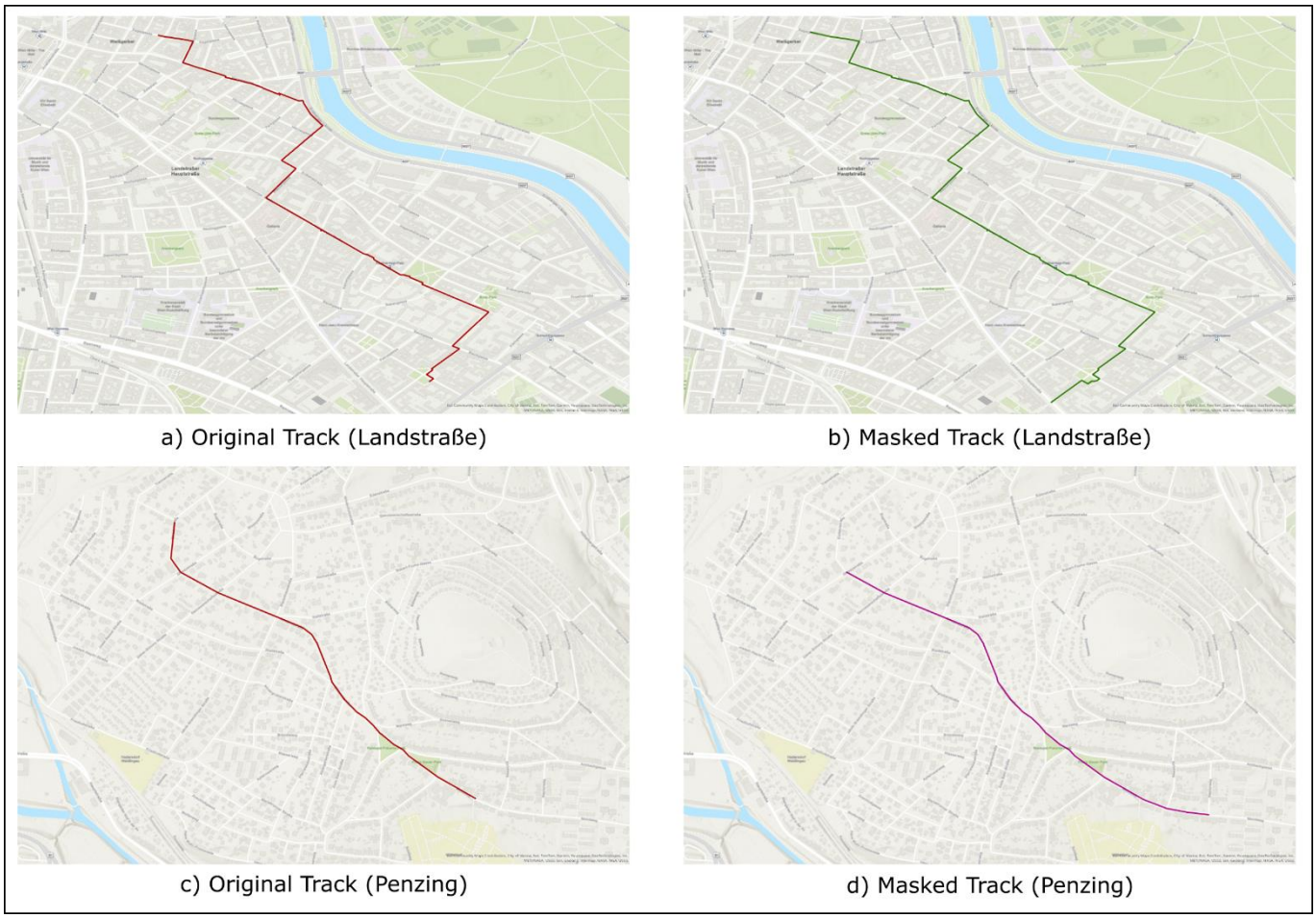


Figure 20: The result of the geomasking method with diverse start and end locations in Landstraße and Penzing, where the route is extended/shortened (k-max = 20)

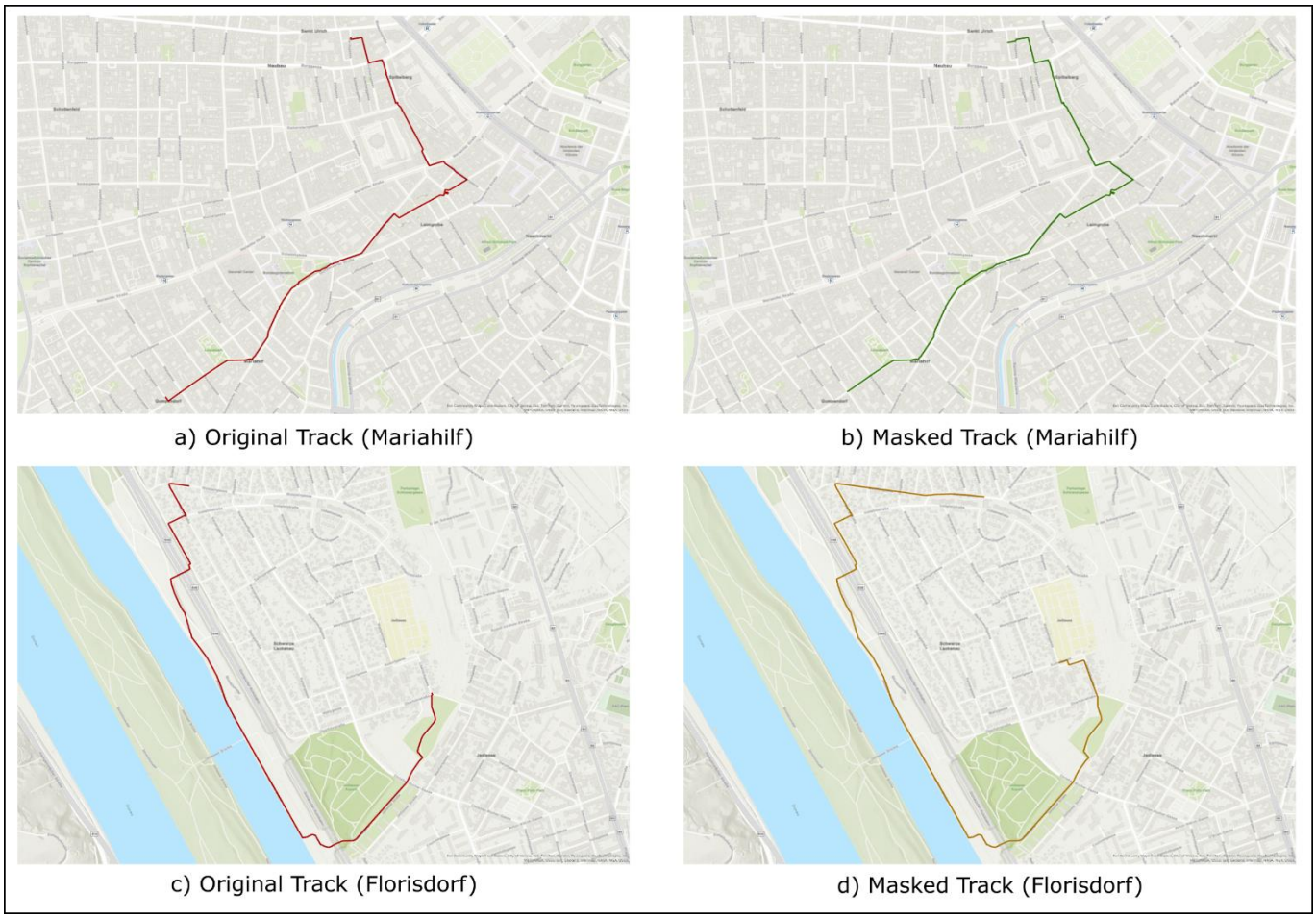


Figure 21: The result of the geomasking method with diverse start and end locations in Mariahilf and Florisdorf, where the route is extended/shortened (k-max = 50)

Next, figures 22-23 depict masked routes, wherein the tracks are slightly modified. Particularly the original start and/or end locations are excluded from these masked routes. This behaviour occurs due to the fact that, during the early stages of the geomasking process, a certain amount of track points are removed from the original route and the masked locations are shifted to another street. This causes the removal of the original start and/or end location from the masked track. Consequently, this outcome also successfully masks the exact start and/or end locations, thereby enhancing the athlete's privacy protection. This makes it significantly more challenging for an attacker to infer the original start and/or end location, thereby reducing the risk of re-identification.

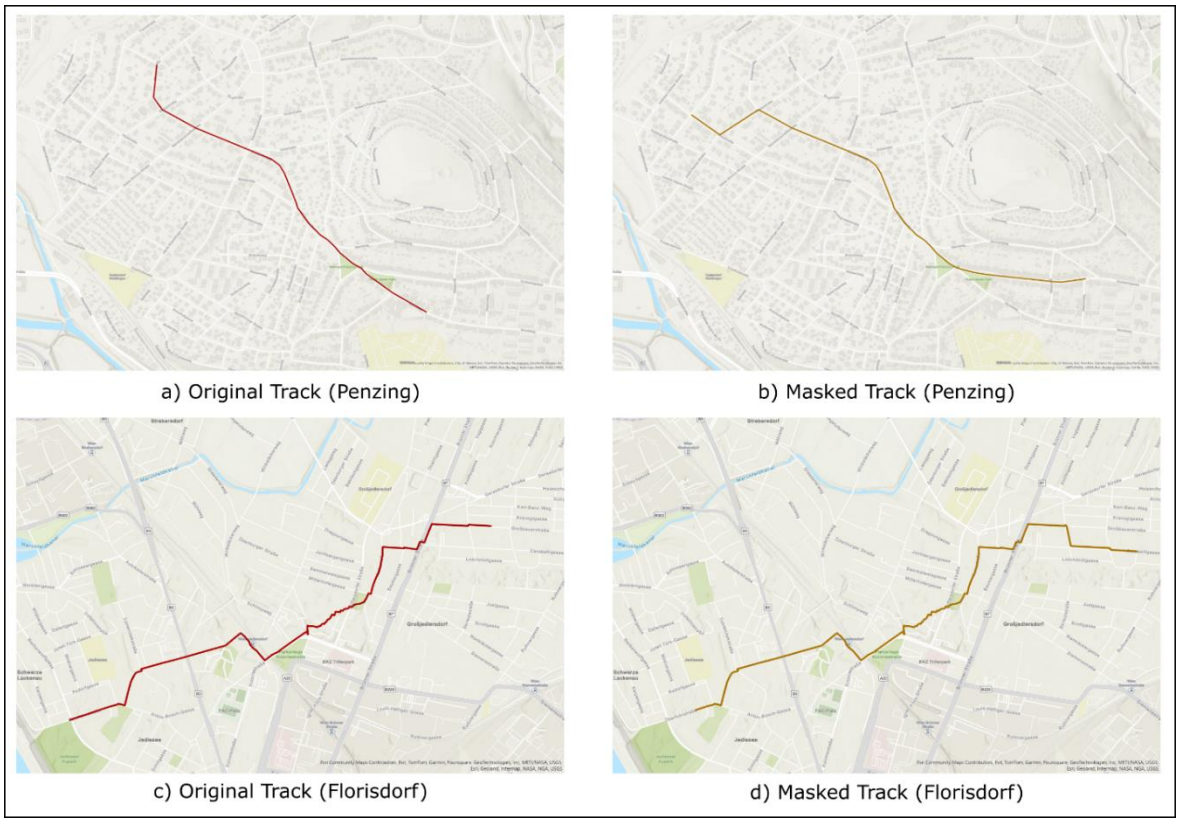


Figure 22: The result of the geomasking method with diverse start and end locations in Penzing and Florisdorf, where the start and end locations are not present in the masked route (k-max = 20)

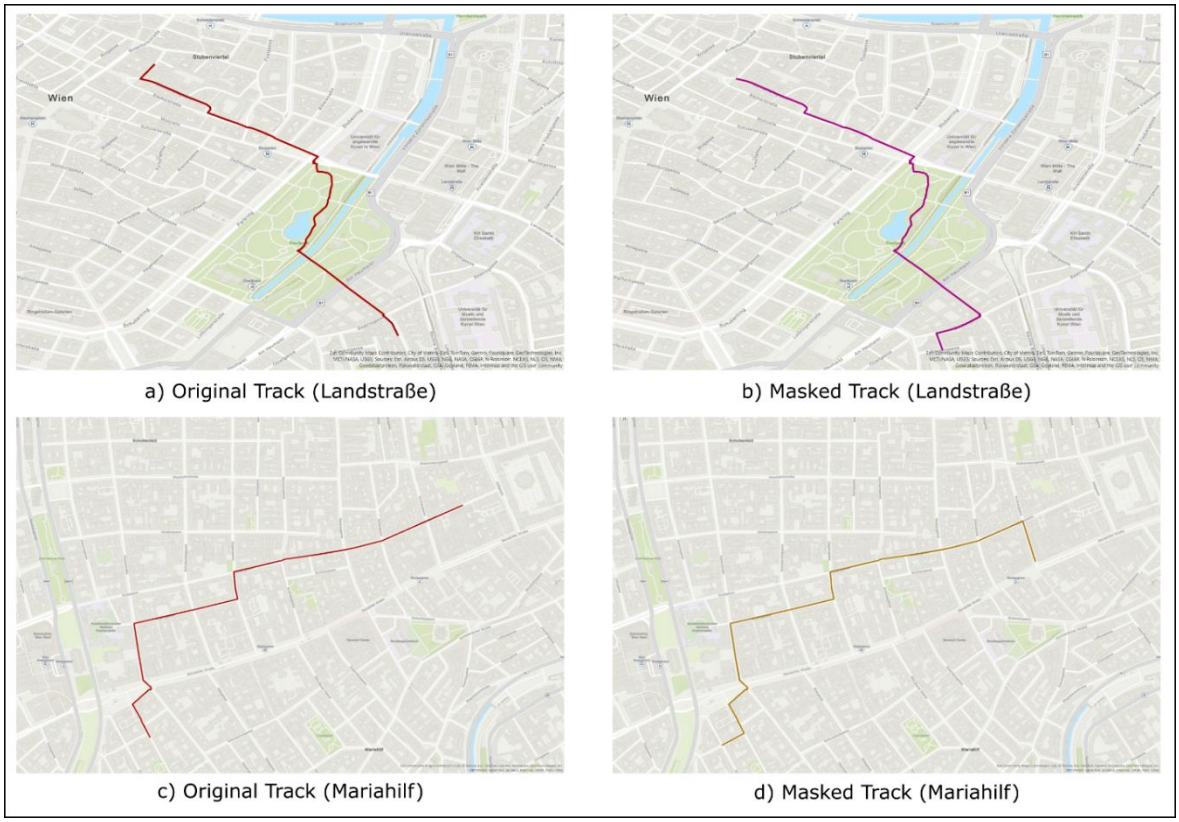


Figure 23: The result of the geomasking method with diverse start and end locations in Landstraße and Mariahilf, where the start and end locations are not present in the masked route (k-max = 50)

Furthermore, it is possible that the original and masked routes are basically identical. This can happen when the start and/or end locations are next to a street intersection. Once the masking method had been applied, the street intersection in question was the closest one to the masked location. The similarity between the original and masked routes does not represent a potential vulnerability to the ska-based geomasking method. Nevertheless, this may prompt users to challenge the efficacy of the geomasking method, particularly given their lack of background knowledge in this area. It is important to note that this behaviour is not a potential risk for an accurate re-identification, as the attacker lacks the necessary information to assess the similarity with the original route. It is crucial to strike a balance between maintaining the unpredictability and efficacy of the privacy protection method.

This behaviour is seen in figure 24. In figure 24a and 24b, both the masked start and end locations were close to the original start and end locations, and this resulted in the masked route almost being identical. In figure 24c and 24d, this behaviour can be seen on one side of the route.

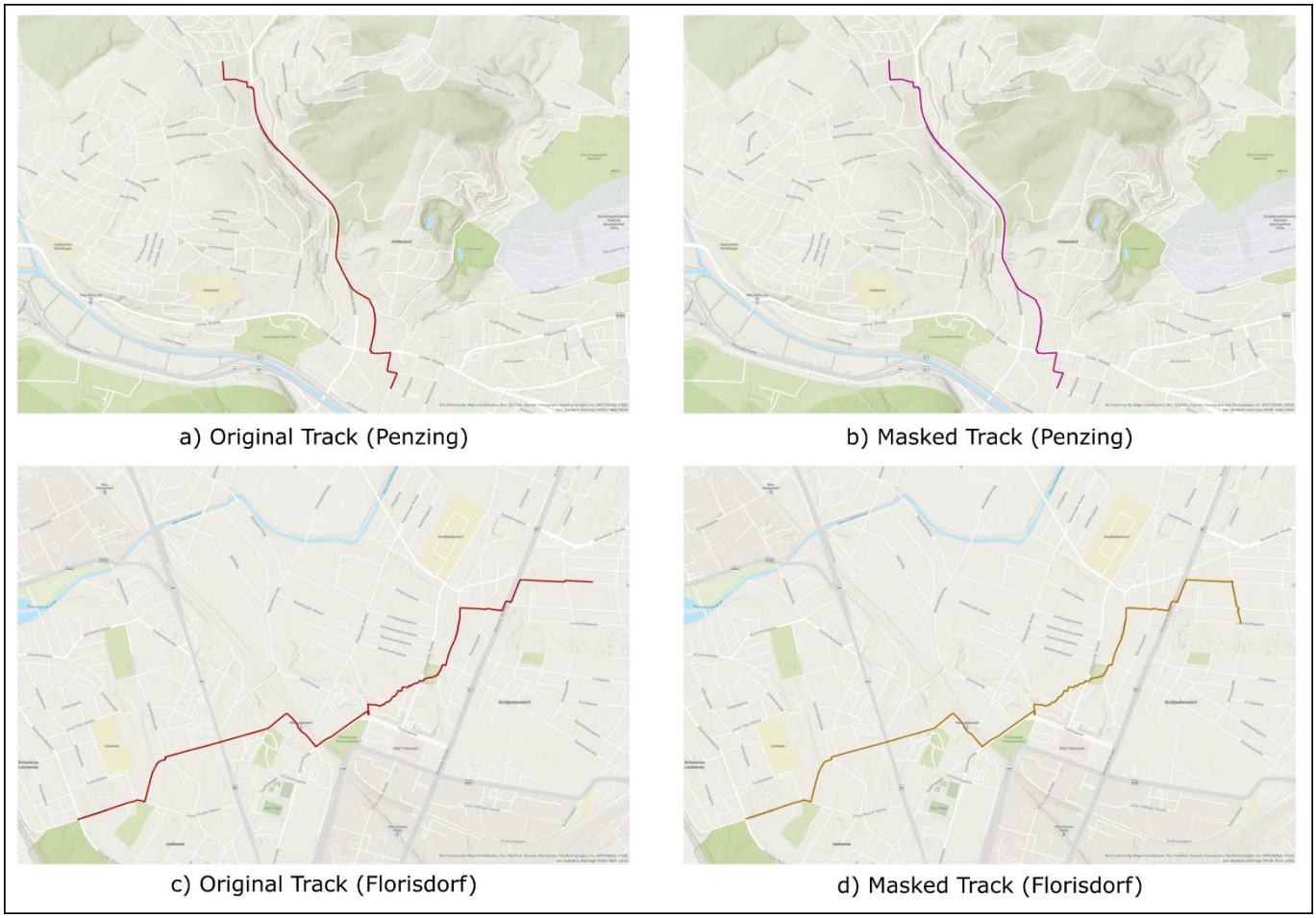


Figure 24: The result of the geomasking method with diverse start and end locations in Penzing (k-max = 20) and Florisdorf (k-max = 50), where the masked route is similar to the original route

The final observation derived from the visual analysis of the results is that an insufficient number of points in the original track were deleted at the beginning of the algorithm. It usually happens when not enough points have been removed to the nearest road intersection from k-max. This results in the generation of a new route which is illogical, such as an unnatural or unexpected turn or creating a path that does not make sense. Furthermore, it is clear that the route was tampered with, and this decreases the efficacy of the ska-based geomasking method.

Figure 25 illustrates this behaviour, whereby a part of the original track can still be seen, and the resulting route has an unexpected turn, which renders the route illogical. This outcome also influences the usability of the user.

Figure 25: The result of the geomasking method with diverse start and end locations in Landstraße (k-max = 20) and Florisdorf (k-max = 20 and 50), where not enough points were deleted

4.1.2 Identical Start and End Location

This subsection presents the findings into the efficacy of the developed ska-based privacy protection method on routes with an identical start and end location. The following results indicated that the developed geomasking method also shows promising results for protecting the sensitive location information in cases where the track has identical start and end points. However, the incorporation of new GPX segments proved ineffective in certain scenarios, thereby effecting the usability for users. Consequently, modifications to the algorithm will be imperative to address these weaknesses.

First, two outcomes were observed, both of which involved modifications to the original track in a manner that excluded the sensitive location from the masked track. These findings illustrate the effectiveness of the geomasking technique in achieving a balance between protecting sensitive location data and maintaining the usability of the route for athletes. Coincidentally, during the testing phase, these results were identified on two occasions, both involving the same route. Figure 26 provides a visual representation of these outcomes, offering a clearer understanding of this behaviour, with the blue points disclosing the original start and end points of the route.

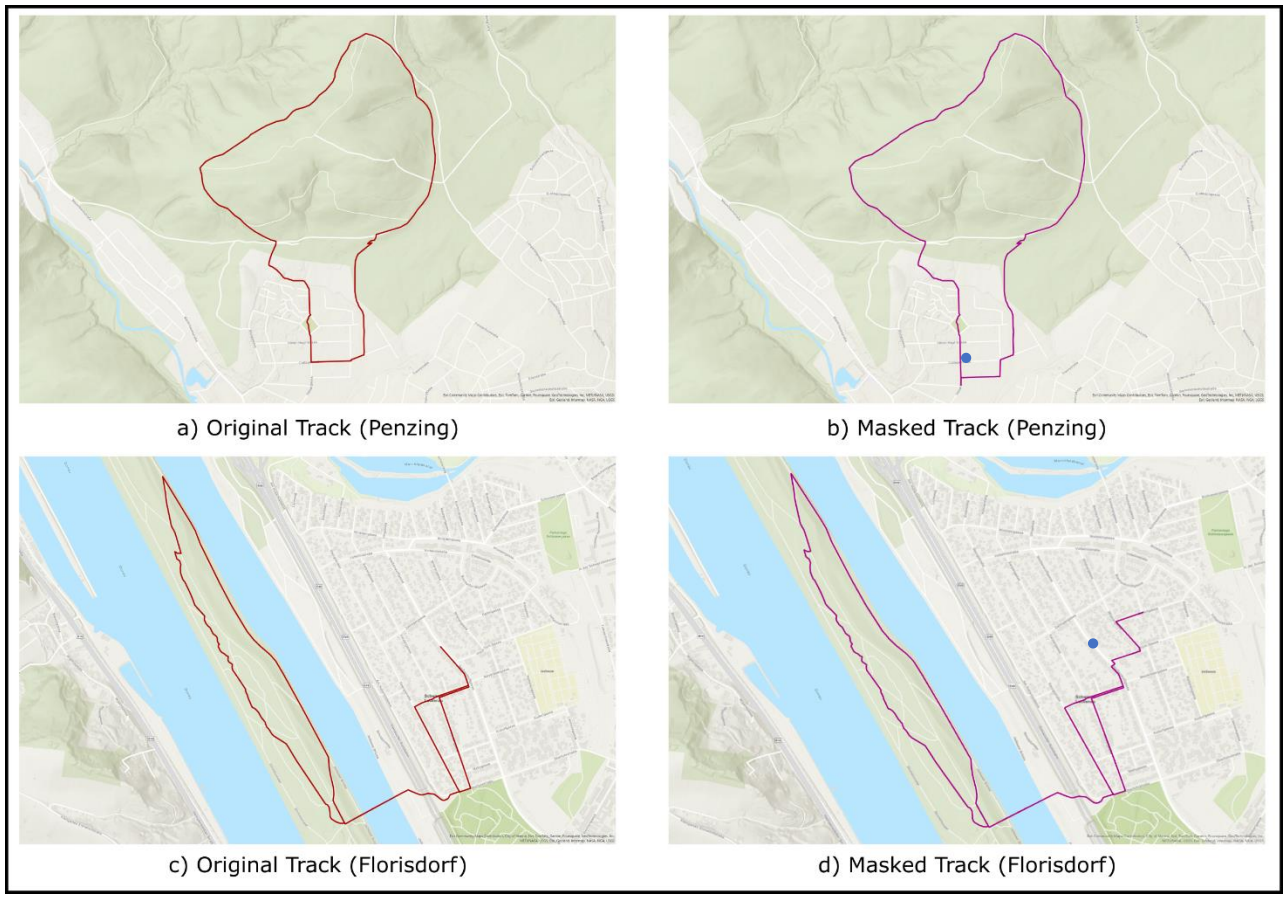


Figure 26: The result of the geomasking method with identical start and end locations in Penzing (k-max = 50) and Florisdorf (k-max = 20), where the sensitive location is excluded in the masked track

Next, there were also instances where the masked track was identical to the original track, depicted in figure 27. This result comes to fruition when both the masked start and end location are displaced to a street intersection that is included in the original track. This behaviour in routes with the same start and end location, like in routes with a different start and end location, does not present a potential vulnerability to the ska-based geomasking method. First of all, it should be noted that an attacker who looks at this route is most likely unable to identify the precise location at which the athlete started and ended their training activity within the closed route. Nevertheless, in the event that an athlete employs a geomasking technique and observes no alterations in the masked route, this may be confusing to the user, potentially affecting the overall usability of the geomasking method.

However, this outcome should be treated with caution. In instances where multiple training activities are recorded with identical start and end locations, resulting in the creation of a closed track, a possible approach to infer the sensitive location data could be to overlay the routes and identify the areas of overlap. Thereby, an attacker can narrow down the possible start and end location.

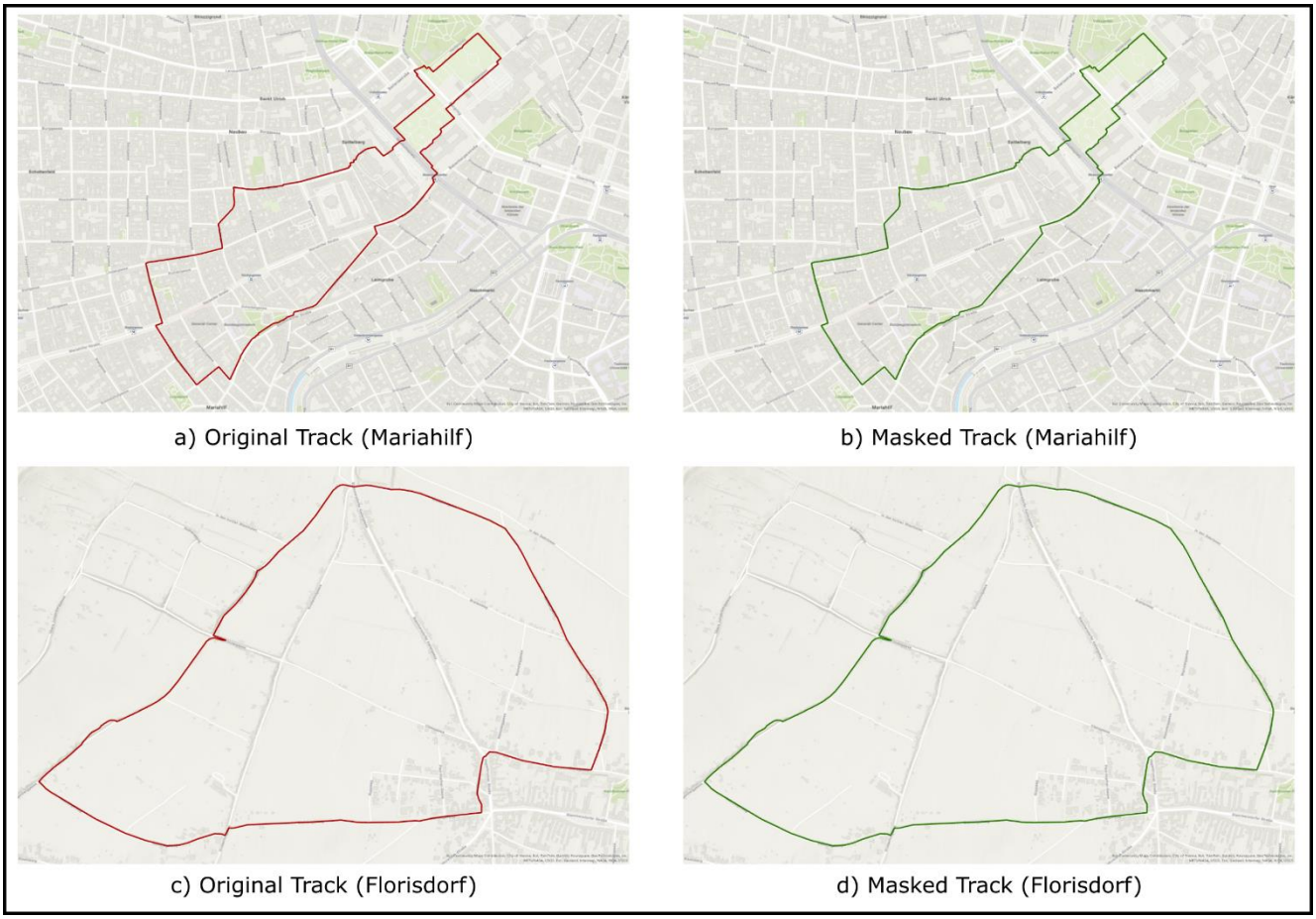


Figure 27: The result of the geomasking method with identical start and end locations in Mariahilf and Florisdorf, where the original and masked tracks are identical (k-max = 20)

Furthermore, there were outcomes where the masked route was slightly changed. The masked start and end points were displaced in a manner that rendered the masked track unconnected, which makes it appear that an EPZ may have been utilised as a privacy protection method. This result is presented in figures 28 and 29 and generally discloses the general area in which the training activity was initiated and concluded. In the event of a gap being present, there are two possible scenarios. Either the sensitive location is not present within the gap, depicted in figure 28, or it can be identified within the gap, which can be seen in figure 29. However, it does not significantly increase the risk of re-identification with a probability higher than what the user-defined spatial k-anonymity offers.

The disclosure of the approximate area where the activity was started and concluded represents a potential weakness in the geomasking method. Subsequently, the algorithm could be enhanced by addressing these gaps and closing the identified gaps or by processing the start and end location once, instead of twice. This results in the behaviour previously described in this subsection, whereby the masked track is closed, and the sensitive location information is excluded, or alternatively, the masked track is identical to the original track.

Figure 28: The result of the geomasking method with identical start and end locations in Landstraße, where the sensitive location is not within the gap (k-max = 20)

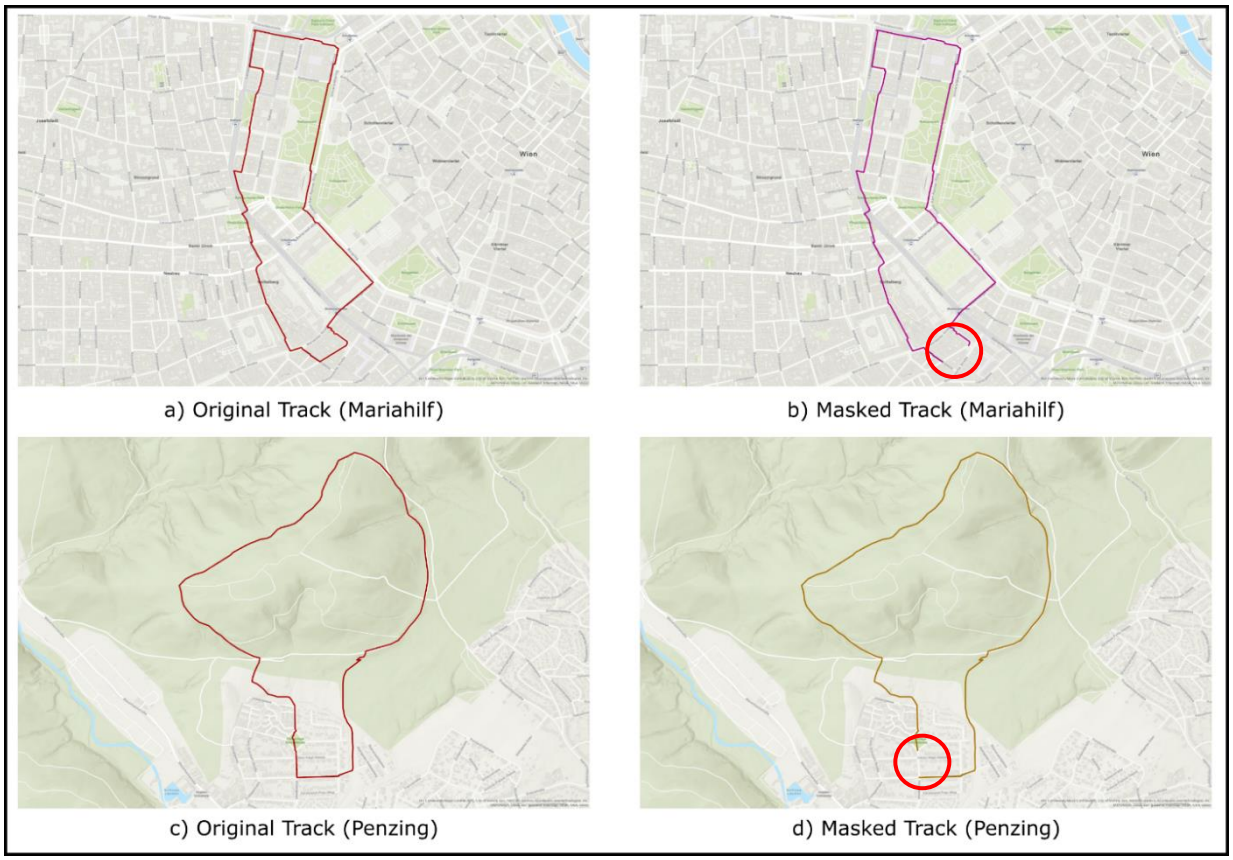


Figure 29: The result of the geomasking method with identical start and end locations in Mariahilf and Penzing, where the sensitive location is within the gap (k-max = 20)

Subsequently, it can also occur that one masked start or end location is relocated to a street intersection which is included in the original route and the other one is relocated to a street junction that is not included in the original track. This happens because both the start and end location, even though they are the same, are currently processed separately in the ska-based geomasking method. Figures 30-31 presents this outcome and it can be seen that the shape of the original track is preserved, but a new segment is added to the masked track and the sensitive location information is still included in the track.

This outcome also shows a good balance between protecting the sensitive location information and providing a good usability for the athletes, as the general shape of the track is preserved, and the track is logical. However, it must be noted that this outcome can also disclose the general area where the training activity was started and ended. However, this can also prove misleading to the attacker, as it is possible that an athlete may have started and concluded their training activity at the end of the new segment. Nevertheless, this behaviour also does not significantly increase the risk of re-identification with a probability higher than what the user-defined spatial k-anonymity offers.

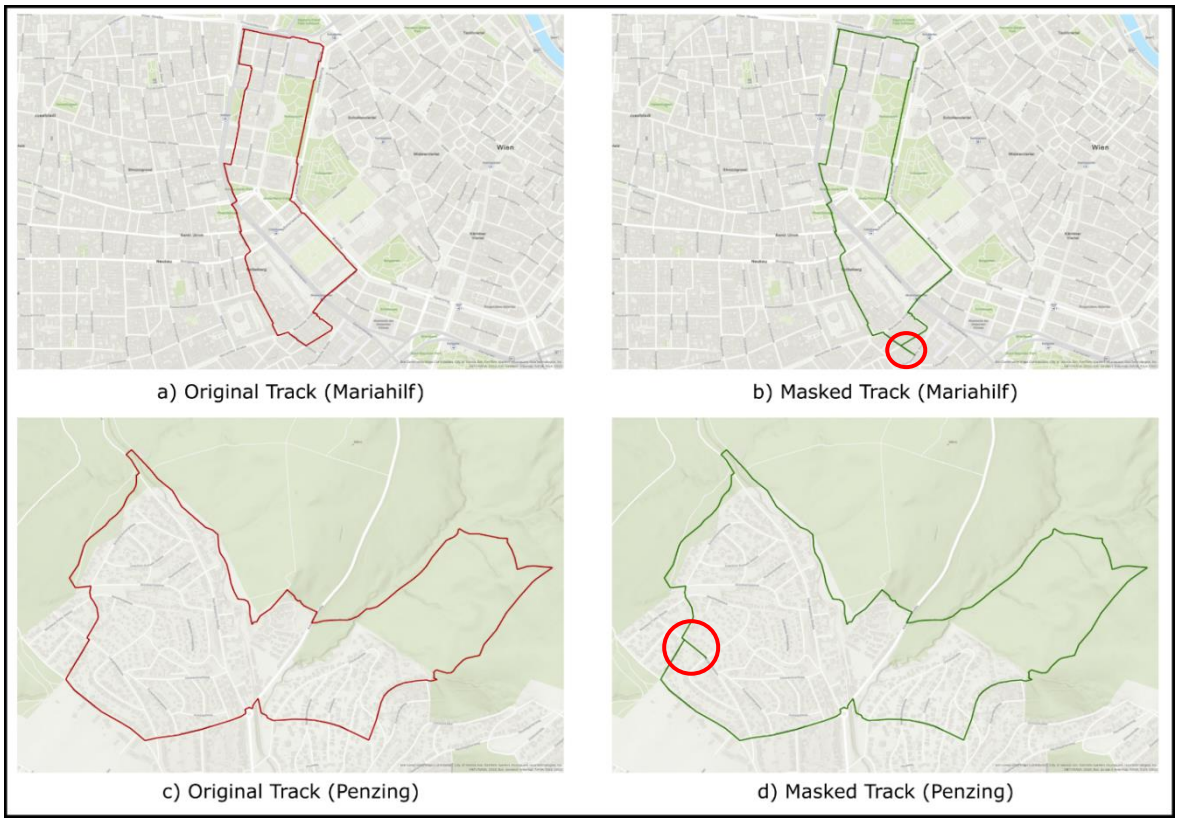


Figure 30: The result of the geomasking method with identical start and end locations in Mariahilf and Penzing, where a segment is added to the original route (k-max = 20)

Figure 31: The result of the geomasking method with identical start and end locations in Landstraße and Florisdorf, where a segment is added to the original route (k-max = 50)

A similar behaviour that happens with tracks which have a different start and end location can also happen with tracks that have the same start and end location. Specifically, that insufficient points are deleted at the beginning of the algorithm, which is illustrated in figure 32. Furthermore, this result makes the track look illogical and demonstrates that the track was modified, which can raise concerns about the usability for the athletes. In certain scenarios, with the background of how the algorithm works, an attacker can possibly narrow down the sensitive location to a specific region in the map or even a particular segment of a street, which represents a weakness. However, this outcome also does not significantly increase the risk of re-identification with a probability higher than what the user-defined spatial k-anonymity offers.

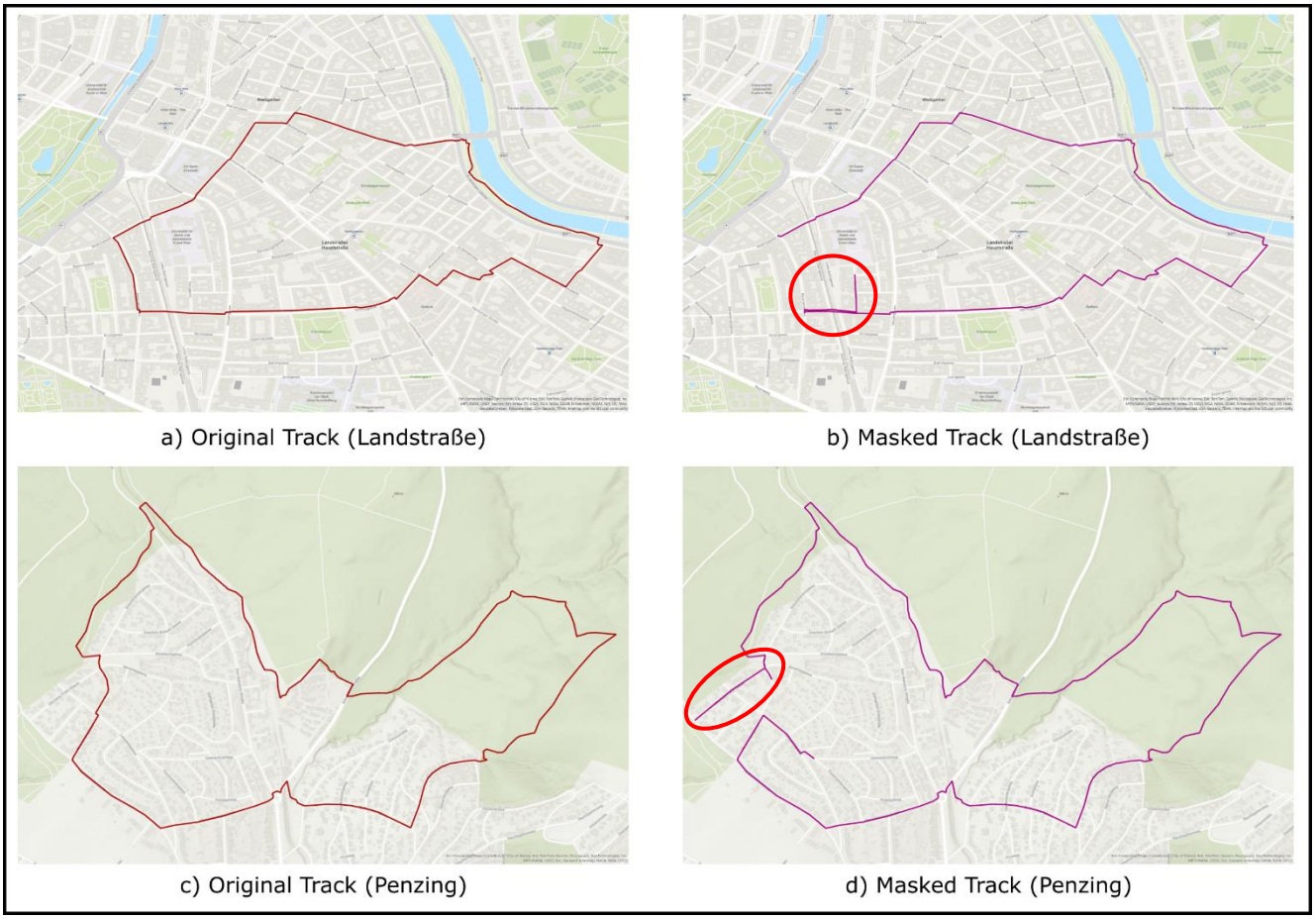


Figure 32: The result of the geomasking method with identical start and end locations in Landstraße (k-max = 50) and Penzing (k-max = 20), where not enough points were deleted

Lastly, there was also one outcome, where the masked route is not logical, and was modified excessively. which influences the usability of the athletes. This result is visualized in figure 33.

Figure 33: The result of the geomasking method with identical start and end locations in Florisdorf, where the masked track is not logical (k-max = 50)

4.1.3 Summary of the Findings

Overall, the results suggest that the developed ska-based privacy protection method, when applied to routes with a different start and end location and the same start and end location, show promising results for protecting the sensitive location information. Nevertheless, some weaknesses were identified, which have the potential to impact the usability for its intended users. A consequence may be that athletes will not utilise the mechanism that is designed to protect their sensitive location information, because of these usability issues, even though an attacker still has a low probability of successfully re-identifying the sensitive location.

There was one weakness identified during the testing of routes with a diverse start and end location, namely that an insufficient number of points are removed at the beginning of the algorithm. This weakness was also observed for routes which have the same start and end location. A possible measure to counter this weakness is to implement a dynamic removal of points from the original route in combination with the user-defined ska-level. The distance between the original point and the furthest possible masked location can be calculated, and a buffer with this distance is drawn around the original location. Subsequently, the points of the original track that fall within the specified buffer area are removed from the track, and the newly identified start and end locations are utilised for the further processing of the algorithm. This approach may help to further enhance the usability aspect of the ska-based privacy protection method, especially for routes with a different start and end location. Nevertheless, this must be thoroughly tested, as new problems could derive, for instance, the masked track is altered to a significant extent. This may be unattractive for athletes to use, as there must be a good balance between the similarity of the original track and providing a protection of sensitive location information.

On the other hand, there are weaknesses with the ska-based geomasking method when applied to routes with the same start and end locations. The main weakness for these kinds of routes is that currently the start and end locations are being processed separately, even though they are the same. A possible way to counter some behaviours observed may be to process such a route differently. For example, the algorithm should check if the start and end location are identical. In such an instance, the location should be masked on a single occasion, rather than twice, as is currently the case. As a result, the masked track should always be connected and should not have gaps anymore. The existence of a gap in a closed track can provide insight into the approximate area where the route was initiated and ended. However, this adaptation of the ska-based privacy protection method must also be rigorously tested to ascertain whether it offers an improvement to the athlete's privacy protection.

Another additional method for addressing the potential weakness of gaps in the masked track, is to implement the detection of such gaps and generate a new GPX segment to effectively close the track again. This can prevent an attacker from estimating where the training activity was started and ended. An alternative approach to addressing this potential issue is to detect any gaps in the masked track, then close them and create a deliberate gap in another section of the track. This additional measure serves to further enhance the protection of the sensitive location information, by directing an attacker's attention away from the sensitive location information.

Furthermore, an additional observation was made regarding tracks with identical start and end points. It is important to consider whether these types of trajectories should be masked, given that an attacker is in most cases unable to ascertain the starting and ending points of an athlete's training activity. However, in the event that a user has multiple different routes tracked, which started and ended at the same location and a geomasking method was not applied, an attacker could overlap the different trajectories and identify, which parts of the tracks overlap. Subsequently, the attacker may direct their attention to these areas of overlap.

It is also important to note that the testing phase of this dissertation may not have fully captured every possible outcome of the ska-based privacy protection technique. Furthermore, urban and suburban areas in Vienna have been used for the examination of the geomasking technique. More testing should be done, also in other urban and suburban areas, to fully understand the performance of the geomasking method. An investigation in rural areas should be conducted as well. Lastly, more levels of ska should be investigated. Nevertheless, as the level of ska increases, the process of generating the masked track becomes increasingly more challenging.

4.2 Exploration of the Mean and Median Distance Displacement

The analysis of the mean and median distance displacement can provide a valuable insight into the geomasking technique. By examining the mean displaced distance, it is possible to investigate the average movement of the locations. Furthermore, the mean is sensitive to outliers within the dataset, which has the potential to result in an inaccurate representation of the distance displacement associated with the geomasking technique. However, the median value is less affected by outliers, which generally provides a better representation of the ska-based privacy protection technique. This is why both metrics are calculated and examined, providing a more comprehensive understanding of the average distance displacement of the geomasking method across the different areas in Vienna.

The results of the mean and median distance displacement from the original start and end location to the masked start and end location are separated by district and are presented in table 8 and 9.

k-max = 20	**Landstraße**	**Mariahilf**	**Penzing**	**Florisdorf**
Mean	71,64 m	68,06 m	131,14 m	195,62 m
Median	60,29 m	71,93 m	137,40 m	179,21 m

Table 8: Mean and Median distance displacement between the original and masked points (k-max= 20)

k-max = 50	**Landstraße**	**Mariahilf**	**Penzing**	**Florisdorf**
Mean	108,67 m	108,64 m	160,62 m	298,33 m
Median	111,41 m	101,24 m	155,28 m	199,10 m

Table 9: Mean and Median distance displacement between the original and masked points (k-max = 50)

Regarding the distance displacement between the various districts, it can be observed that the distance displacement is greater in the outer districts (suburban areas) than in the inner districts (urban areas). This indicates that the masked locations are displaced further away because the population and building density is generally lower in suburban areas. These results obtained are in accordance with the anticipated outcomes. The mean and median values at both levels of ska and across the four different districts are found to be relatively similar. However, in the district of Florisdorf, at the ska level of 50, the highest difference is observed, indicating the presence of a few outliers.

The following figures illustrate examples of the displacement of the original location to the masked locations from the ska-based geomasking technique. These visualizations demonstrate the extent and variability of the displacement of the masked points across different routes.

Figure 34: Displacement of the start and end location in Landstraße (k-max = 20)

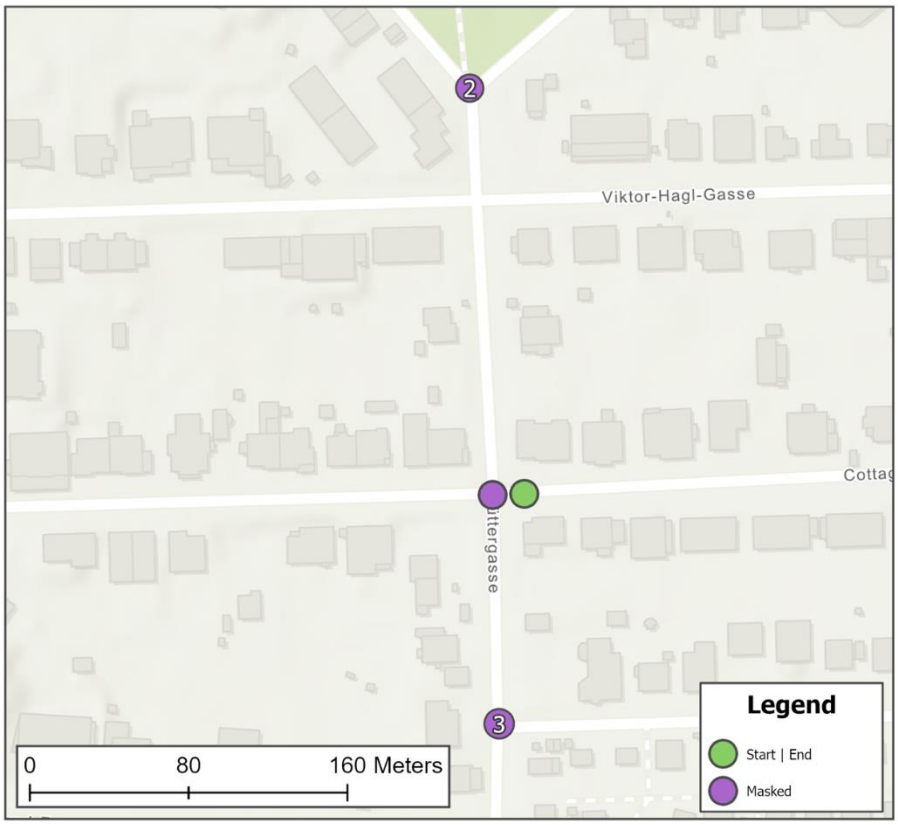


Figure 35: Displacement of the start and end location in Penzing (k-max = 20)

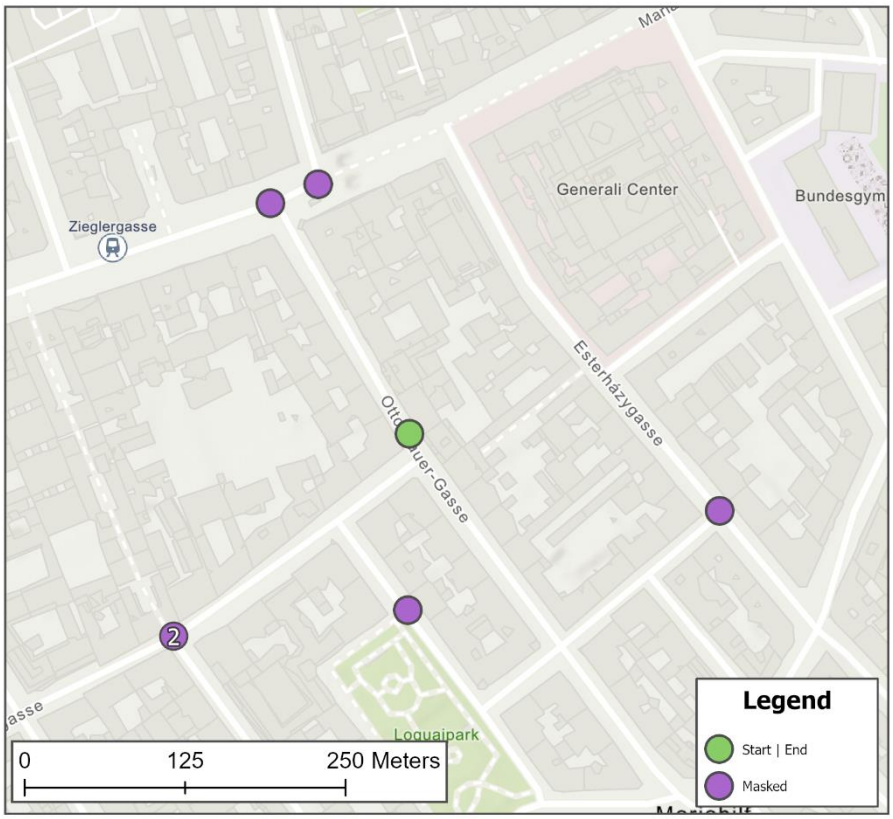


Figure 36: Displacement of the start and end location in Mariahilf (k-max = 50)

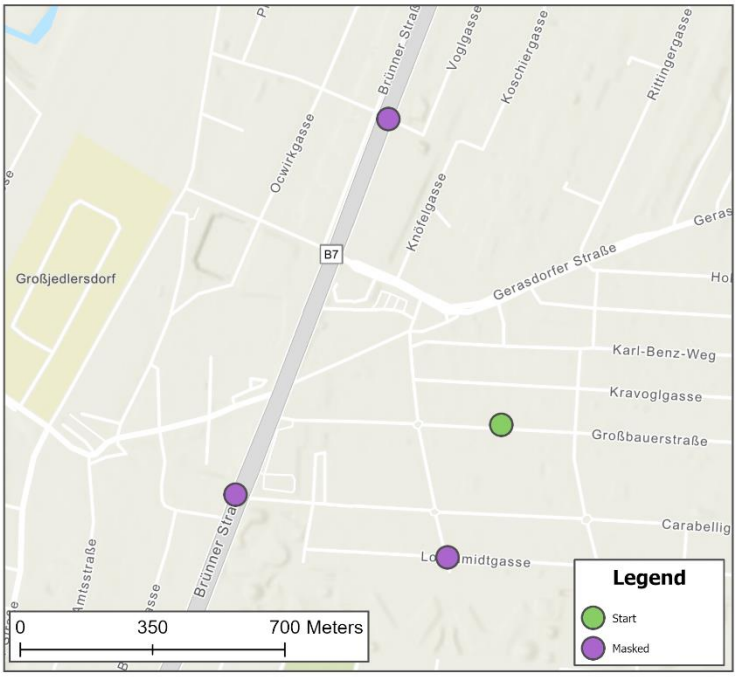


Figure 37: Displacement of the start and end location in Florisdorf (k-max = 50)

4.3 Recommendation to Handle the Travelled Distance and Time Metrics

After modifications, such as the use of a geomasking method, have been applied to a tracked route, the total distance travelled has likely changed. When a geomasking technique that masks the original location by distance, and the travelled distance is kept and shared, it presents a significant risk to the user's sensitive privacy information. However, the geomasking method developed and examined in this master thesis does not directly mask the sensitive location information by distance but with addresses and street intersections. Because of this approach, the distance displacement of the masked location varies.

In tables 10 to 13 the original travelled distance and the masked travelled distance are presented. For a better overview, the tables are organized according to the two distinct routes and ska levels. Routes 1 and 2 are characterized by a different start and end location, whereas routes 3 and 4 have the same start and end location.

District	Route	Iteration	Original Distance (km)	Masked Distance (km)	Difference (km)	Average Difference (km)
Landstraße	Route 1	1	1,37	1,36	-0,01	0,01
		2		1,32	-0,05	
		3		1,46	0,09	
	Route 2	1	2,59	2,83	0,24	0,17
		2		2,71	0,12	
		3		2,75	0,16	
Mariahilf	Route 1	1	1,45	1,61	0,16	0,24
		2		1,75	0,30	
		3		1,70	0,25	
	Route 2	1	2,24	2,29	0,05	0,04
		2		2,23	-0,01	
		3		2,32	0,08	
Penzing	Route 1	1	1,22	1,43	0,21	0,16
		2		1,25	0,03	
		3		1,46	0,24	
	Route 2	1	2,65	2,79	0,14	0,08
		2		2,66	0,01	
		3		2,75	0,10	
Florisdorf	Route 1	1	3,84	4,25	0,41	0,41
		2		4,27	0,43	
		3		4,22	0,38	
	Route 2	1	2,78	2,91	0,13	0,40
		2		3,28	0,50	
		3		3,36	0,58	

Table 10: Original and Masked travelled distance for routes with a different start and end location (k-max = 20)

District	Route	Iteration	Original Distance (km)	Masked Distance (km)	Difference (km)	Average Difference (km)
Landstraße	Route 1	1	1,37	1,36	-0,01	0,07
		2		1,41	0,04	
		3		1,55	0,18	
	Route 2	1	2,59	2,66	0,07	0,07
		2		2,72	0,13	
		3		2,59	0	
Mariahilf	Route 1	1	1,45	1,40	-0,05	0,12
		2		1,78	0,33	
		3		1,52	0,07	
	Route 2	1	2,24	2,20	-0,04	0,02
		2		2,44	0,20	
		3		2,13	-0,11	
Penzing	Route 1	1	1,22	1,14	-0,08	0,05
		2		1,47	0,25	
		3		1,21	-0,01	
	Route 2	1	2,65	2,90	0,25	0,16
		2		2,68	0,03	
		3		2,85	0,20	
Florisdorf	Route 1	1	3,84	4,41	0,57	0,41
		2		4,40	0,56	
		3		3,94	0,10	
	Route 2	1	2,78	3,11	0,33	0,54
		2		3,29	0,51	
		3		3,55	0,77	

Table 11: Original and Masked travelled distance for routes with a different start and end location (k-max 50)

District	Route	Iteration	Original Distance (km)	Masked Distance (km)	Difference (km)	Average Difference (km)
Landstraße	Route 1	1	1,37	1,36	-0,01	0,07
		2		1,41	0,04	
		3		1,55	0,18	
	Route 2	1	2,59	2,66	0,07	0,07
		2		2,72	0,13	
		3		2,59	0	
Mariahilf	Route 1	1	1,45	1,40	-0,05	0,12
		2		1,78	0,33	
		3		1,52	0,07	
	Route 2	1	2,24	2,20	-0,04	0,02
		2		2,44	0,20	
		3		2,13	-0,11	

Penzing	Route 1	1	1,22	1,14	-0,08	0,05
		2		1,47	0,25	
		3		1,21	-0,01	
	Route 2	1	2,65	2,90	0,25	0,16
		2		2,68	0,03	
		3		2,85	0,20	
Florisdorf	Route 1	1	3,84	4,41	0,57	0,41
		2		4,40	0,56	
		3		3,94	0,10	
	Route 2	1	2,78	3,11	0,33	0,54
		2		3,29	0,51	
		3		3,55	0,77	

Table 12: Original and Masked travelled distance for routes with the same start and end location (k-max = 20)

District	Route	Iteration	Original Distance (km)	Masked Distance (km)	Difference (km)	Average Difference (km)
Landstraße	Route 3	1	4,09	4,22	0,13	0,11
		2		4,08	-0,01	
		3		4,29	0,2	
	Route 4	1	3,60	3,57	-0,03	0,16
		2		3,77	0,17	
		3		3,93	0,33	
Mariahilf	Route 3	1	4,74	4,94	0,20	0,13
		2		5,01	0,27	
		3		4,66	-0,08	
	Route 4	1	4,12	3,89	-0,23	0,04
		2		4,26	0,14	
		3		4,34	0,22	
Penzing	Route 3	1	4,80	4,80	0	0,09
		2		4,99	0,19	
		3		4,88	0,08	
	Route 4	1	5,42	5,53	0,11	0,07
		2		5,44	0,02	
		3		5,51	0,09	
Florisdorf	Route 3	1	4,46	5,74	1,28	0,94
		2		5,34	0,88	
		3		5,12	0,66	
	Route 4	1	5,82	6,55	0,73	0,18
		2		5,42	-0,40	
		3		6,03	0,21	

Table 13: Original and Masked travelled distance for routes with the same start and end location (k-max = 50)

It can be observed that in most cases the total distance travelled from the masked route is greater than the original distance travelled. Subsequently, the difference between the original and masked routes varies for practically each iteration from every route. Furthermore, for routes with a different start and end location the ska-based geomasking shows promising results and an attacker is unable to identify if a part of the route was either extended, shortened or did not significantly change. Therefore, it can be argued that it would be possible to share the original distance travelled with the masked track. Nevertheless, to avoid a potential attack on the ska-based privacy protection method the following recommendation is put forth: when a masked track is shared on a fitness tracking application, the total distance travelled from the masked route is also shared. As soon as a certain number of activities are recorded and the tracked routes differ from one another, the original distance travelled from the activities can be aggregated. Thereby, an athlete would be able to participate in the various competitions and leaderboards on the fitness tracking application.

In the current state of the developed geomasking method for routes with the same start and end location the distance travelled can be significantly higher than the actual distance travelled, even though the trajectory only has minimal visual changes. This is due to the fact that one or both of the masked locations are being moved to a street intersection that was incorporated in the original route already. Consequently, the total distance travelled in the masked route is greater, as this section of the route is counted twice. However, this issue can be negated by processing the start and end location only once, as opposed to separately as previously discussed. This ensures that there would be no overlapping segments in the trajectory anymore.

Furthermore, in the course of the testing of the routes, the masked route was found to be longer than the original route in 80% of cases. The average distance travelled at k-max of 20 is 0,15 km, while at k-max of 50, it is slightly higher at 0,20 km. Moreover, the range (maximum to minimum value) of the differences for each k-max value was calculated. At k-max of 20, the range amounts to 0,72 km and at k-max of 50, the range is 1,68 km. The findings indicate the presence of outliers in the test results. Lastly, the standard deviation of the difference between the original and masked travelled distance is calculated. The standard deviation shows how spread out the data relative to the mean of the dataset is. The standard deviation at k-max of 20 is 0,16 km, while the standard deviation at k-max of 50 amounts to 0,30 km.

Average difference k-max = 20 (km)	Range of difference k-max = 20 (km)	Standard deviation of k-max = 20 (km)	Average difference k-max = 50 (km)	Range of difference k-max = 50 (km)	Standard deviation of k-max = 50 (km)
0,15	0,72	0,16	0,20	1,68	0,30

Table 14: The average, range and standard deviation of the difference in the total distance travelled between the original and masked routes

These results, which are also presented in table 14, demonstrate a notable degree of variability in the total travelled distance for the trajectories, as evidenced by the higher standard deviation values in comparison to the mean (average) values. This indicates that there is not only a substantial degree of variability in the dataset, but also that the data points are distributed across a wide range of values, some of which may be considerably different compared to the mean. This is also evident from the range value.

Additionally, the difference between the total distance travelled from the original and masked routes are greater in the suburban areas. This is due to the fact that the population and building density are lower in these areas, and the road network in suburban areas is typically distinct from that of urban areas. Furthermore, the proximity of the start and end points to street intersection also has an influence. It can be observed that start and end points situated at a greater distance from a street intersection will typically have a greater change in the total distance travelled.

It is nevertheless important to note that an underestimation or overestimation of the route makes it unattractive for the athlete and renders a geomasking method unsuitable. In such cases, it is necessary to strike a balance between the usability of the method and the protection of sensitive locations.

Lastly, the data collected from running tracks is spatio-temporal. The summary of the training activity not only records the distance travelled, but also incorporates time metrics. The total time and pace (time per kilometre) is normally tracked as well. The total time should also be adapted to the total travelled distance. This can be achieved by taking the pace of the training activity and modifying the total time according to the masked travelled distance.

5 Conclusion

The goal of this thesis was to develop a ska-based privacy protection method for fitness tracking applications based on masking the sensitive location information with nearby addresses and street intersections, to enhance the individual's privacy while preserving the usability of the route. First, the research questions posed at the beginning of this study are answered. Then, some limitations during the thesis are explained in the next subsection. Lastly, some recommendations for future research and adaptations for the ska-based privacy protection technique based on the results of this study are given.

5.1 Answering the Research Questions

Research question 1: *Which protective mechanisms do mobile fitness applications offer?*

In this study a total of ten mobile fitness tracking applications were observed and 3 main protective mechanisms were found to be integrated across these different fitness applications. First, individuals have the ability to put their profile and fitness activities to private. This results in other athletes on the fitness tracking application to not see their activities. Furthermore, it is also possible to share training activities with only friends on the social platforms. However, other individuals are still able to see your profile with limited details. Another protective mechanism available in these mobile applications is the possibility to block other individuals. While this feature can have different effects in fitness tracking applications, it generally prevents the blocked user from seeing any fitness activities. Lastly, a few mobile fitness tracking applications offer a protective mechanism called endpoint privacy zone. This privacy protection technique lets an individual define an area, and if the tracked route starts or ends inside this area, the section until the border of the privacy zone is hidden from other athletes. Additionally, the user is able to modify the radius of the endpoint privacy zone. The minimum and maximum radius, often with fixed intervals, are defined by the mobile fitness application.

Furthermore, the endpoint privacy zone geomasking method has undergone numerous enhancements over time, with the objective of enhancing its efficacy and usability for users. However, these changes were different for each mobile fitness tracking application. The configuration of the privacy zone varies between applications. While some applications offer alternative shapes, others have retained the conventional circular shape. Providing alternative shapes makes the areas less identifiable and predictable. Moreover, different radii and intervals for the privacy zones are offered by these applications. An additional alteration has been implemented, whereby a random quantity of fuzz (noise addition) is incorporated at the periphery of the endpoint privacy zones. Additionally, spatial cloaking was also added to the method, where the centre of the privacy zones is shifted randomly. However, it is also possible that the endpoint privacy zone protection method is not implemented by some mobile fitness

tracking applications, which can be seen in table 3. In this table, the results from the 10 different mobile fitness tracking applications are presented. Six from the 10 applications were found to provide users with the endpoint privacy zone protection mechanism. Another interesting fact is that one of the most popular applications from Adidas, Runtastic which has over 50 million downloads on the google play store does not provide the endpoint privacy zone protection method.

Nevertheless, the range of options for the protection of sensitive user location information in mobile fitness tracking applications is limited, with only one geomasking method currently available.

Research question 2: *What are the limitations of the existing protective mechanisms by such apps in terms of usage and/or risk of re-identification?*

One limitation of using these existing protective mechanisms in mobile fitness tracking applications is that it limits the social aspect of the applications, as private activities and activities only shared with friends do not contribute towards some achievements, challenges and leaderboards hosted by fitness tracking applications. This motivates the users to share their activities publicly, so that they can take part in these challenges. Research has shown that in the early stages of the EPZ technique it had vulnerabilities. As soon as multiple routes with a different intersected entry and/or exit point in the privacy zone are recorded, the defined radius of the EPZ can be identified by an attacker. However, a successful identification of the radius of an EPZ decreased when an individual used a bigger radius. After some adaptations were made to this geomasking technique, another vulnerability was identified. The total distance travelled was not changed after adaptations were made to the training activity. Despite the alterations to the route's visual appearance, the total distance travelled from the original track was shared in the statistics of the route. Therefore, an attack to infer the sensitive locations was based on the length of the hidden paths within the defined EPZ with the help of a street network dataset. This interference attack was successful in re-identifying the sensitive locations from individuals. Nevertheless, some of these vulnerabilities can be solved with some countermeasures, such as rounding the total distance travelled up or down or adding a random noise to it. However, these countermeasures are accompanied by a decline in the usability of the athletes.

Research question 3: *Can existing geographical masking methods, originally designed for discrete location data, be applied to spatio-temporal trajectories of individuals to protect their sensitive locations?*

In this study multiple existing and established geographical masking techniques were described and demonstrated. These geomasking methods have the potential to be applied to spatio-temporal trajectories of individuals to protect their sensitive locations. However, applying these

geomasking techniques to trajectories in mobile fitness tracking applications may face multiple challenges. The primary challenge is to find a balance between protecting the sensitive location information from an individual and maintaining the integrity of the trajectory. It is also important to note that combining different geomasking methods may possibly offer a better balance. Furthermore, spatio-temporal data includes information about time and distance. This also must be taken into account when applying established geomasking techniques for trajectories and ways to handle and process these metrics must also be reviewed.

Research question 4: *How can the privacy measure of spatial k-anonymity (ska) be applied to individual trajectories and prevent inference attacks of the individual sensitive locations?*

The developed privacy protection technique in this study uses the privacy measure of spatial k-anonymity to protect an individual's sensitive location information in trajectories. The ska-based privacy protection mechanism includes multiple steps for the displacement and adaptation of the trajectory: First, the file in which the trajectory is saved, is taken as an input and the sensitive location data is extracted for further processing. Furthermore, a number of points are removed at the beginning and end of the trajectory, thereby deleting a part of the trajectory. Next, the user defines the desired level of ska. Afterwards, the user-defined ska values are utilised to obtain the nearest neighbour addresses from the sensitive locations and a masked location is selected. Next, the closest street intersections from the masked location are acquired and the closest one is selected. Lastly, a new segment is generated from the nearest street intersection of the masked location to the first point of the temporary trajectory. This process is repeated for the end location and the masked trajectory is saved. The objective is to provide a balance between the protection of the athletes' privacy and the usability of the masked trajectory, with the aim of enhancing the appeal and encouraging the use of a privacy protection method in mobile fitness tracking applications.

The ska-based geomasking technique offers four advantages over the current privacy protection method, endpoint privacy zone, offered by these applications. First, the user is able to determine the level of ska. Secondly, the utilisation of address data rather than displacing sensitive locations by distance prevents the masked location from being relocated to an invalid location, such as water bodies or within forests. Thirdly, an additional step is incorporated into the algorithmic process whereby the masked address is selected and subsequently relocated to the nearest street intersection. This serves to further enhance the protection of the data, given that a street intersection typically corresponds to more than one address, with an average of two to four addresses per intersection. This step also reduces the risk of false re-identification. Lastly, a new segment is added to the route which prevents an attacker from guessing where the original start and end points are. As a result of the last two steps, the level of ska can be higher than the user-defined level.

The results from the ska-based geomasking approach showed promising results for protecting the sensitive location information. Furthermore, the mean and median distance displacement was examined. The results were as expected, and the masked locations were displaced further in suburban areas than in urban areas with the same level of ska. Lastly, some recommendations were provided on how to deal with the total distance travelled and the temporal aspects of the trajectories.

In conclusion, the privacy measure of spatial k-anonymity has the potential to be applied successfully to individual trajectories of athletes to prevent inference attacks on the sensitive location information.

5.2 Limitations of the Study

The present study was subject to a number of limitations. First, are some limitations from the open-source data of OpenStreetMap used in the proposed geomasking technique. The accuracy of the data may vary, because it is primarily generated by the general public and OSM contributors. Moreover, it should be noted that in certain geographical areas the open-source data may be of a relatively low accuracy. The initial objective was to solely utilise residential addresses for the identification of the nearest neighbour addresses from the sensitive locations. However, it was found that the classification for residential addresses of the open-source data was very inaccurate and for this reason all addresses were used in order to select the nearest neighbour addresses. Furthermore, it was observed that a single address was sometimes included multiple times in a dataset with slightly different coordinates. A measure was introduced in the algorithm to counter this behaviour. It is nevertheless possible that the nearest neighbour address list contains two entries for the same address.

Another limitation from the open-source data from OpenStreetMap was the identification of the nearest street junctions. The code of the developed geomasking method incorporates a variety of street classifications. However, it is plausible that some streets may be absent or incorrectly classified, as there are dozens of different street classifications in the data. Consequently, a street intersection may be identified as the closest street intersection, yet in actuality, it is in fact situated at a greater distance than the actual closest street intersection, which was determined to be incorrectly identified.

In order to maintain focus on the objectives of this study, a qualitative review has been conducted on a limited number of simulated trajectories in urban and suburban areas in Vienna. Still, many other urban and suburban areas with different characteristics exist, which can be examined. Furthermore, no trajectories in rural areas have been examined by the proposed ska-based privacy protection mechanism. Therefore, it would be of significant importance to apply the developed geomasking method on more trajectories in different geographical areas.

However, there were some weaknesses identified in the geomasking method, particularly when the start and end locations in a trajectory were identical and a different approach to these kinds of routes is suggested.

One algorithmic weakness identified during the testing is that an insufficient number of points are removed at the beginning of the algorithm. This weakness was observed in routes with different and the same start and end locations. In trajectories with the same start and end locations the masked trajectory may appear illogical and significantly different from the original trajectory, which is unattractive to an athlete. Another observed weakness is that certain outcomes, such as a gap in the track, can provide insight into the approximate area where the route was initiated and ended. Lastly, in the current state of the ska-based privacy protection method, the total distance travelled from the masked route in routes with the same start and end points can be higher, despite the trajectory appearing to be visually identical or only slightly changed.

5.3 Future Research Recommendations

The limitations of this work opens up the possibility for future research in the field of protecting sensitive location information in mobile fitness tracking applications. First, it is recommended that the weaknesses of the proposed ska-based privacy protection technique are addressed and that a solution is implemented. A solution for the behaviour that not enough points are being deleted in a trajectory may be to implement a dynamic removal of track points in combination with the user-defined ska-level. The distance between the original (sensitive) location and the furthest possible masked location can be calculated and a buffer with this distance is drawn around the sensitive location. Next, all the points of the original track, which lie inside the buffer areas are removed and the “new” start and end locations are used in the algorithm. This may prove an effective solution. However, it is essential to conduct comprehensive testing to ensure that no new problems arise, such as the trajectory being altered to a significant extent.

Additionally, a different approach for routes with an identical start and end location is suggested. Currently the start and end locations are being processed separately, which can lead to the generation of routes that are significantly different from the original route. Furthermore, there are instances, in which the total distance travelled is considerably higher, even though the route was visually not altered. A potential solution to this problem is to check if the start and end locations are identical or near to each other. If this is the case, then the location should be masked once. This results in the masked track being connected and no gaps should be present anymore. Furthermore, the total distance travelled is more accurate, as a masked location will not be included in the calculation twice, when it is relocated to a street intersection, which is already included in the original trajectory. Nevertheless, this adaptation to the proposed ska-based geomasking method must be rigorously tested to ascertain whether it offers an improvement or if new problems arise.

A further enhancement to the ska-based algorithm may be to introduce a certain degree of noise to the user-defined level of ska. If an athlete utilises the same level of ska on each trajectory, this could potentially lead to a higher chance of re-identification, as an attacker may identify repetitive patterns. The incorporation of a certain degree of fuzziness into the defined level of ska would serve to negate this.

As mentioned above, more testing in other urban and suburban areas and additional testing of the performance of the proposed ska-based privacy protection method in rural areas must be conducted. Nevertheless, further research is required to further develop the proposed ska-based technique and to investigate the utilization of additional geomasking methods for mobile fitness tracking applications. The objective is to identify a solution that offers an optimal balance between the protection of sensitive location data and the usability of the masked trajectory. As it stands today, there are only a very limited number of privacy protection methods available on mobile fitness tracking applications and only one geomasking technique.

It is also imperative that mobile fitness tracking applications must raise the awareness of the individuals about the types of information, including potentially sensitive data, that can be shared with training activities. As evidenced in numerous academic studies (Alrayes & Abdelmoty, 2017; Alrayes et al., 2020; Mink et al., 2022; Zhang & McKenzie, 2023), the general awareness of social media users with regard to the sharing of their data, including that collected through mobile fitness tracking applications, is deficient. It would be optimal to raise the awareness of the users to change the publishing behaviour on social network applications and additionally provide geomasking methods as a means of protecting the sensitive location information.

Bibliography

Agnellutti, C. (2014). Big data: an exploration of opportunities, values, and privacy issues. In *(No Title)*.

Alrayes, F., & Abdelmoty, A. I. (2017). Towards understanding location privacy awareness on geo-social networks. *ISPRS international journal of geo-information*, *6*. https://doi.org/10.3390/ijgi6040109

Alrayes, F. S., Abdelmoty, A. I., El-Geresy, W. B., & Theodorakopoulos, G. (2020). Modelling perceived risks to personal privacy from location disclosure on online social networks. *International journal of geographical information science : IJGIS*, *34*, 176. https://doi.org/10.1080/13658816.2019.1654109

Armstrong, M. P., Rushton, G., & Zimmerman, D. L. (1999). Geographically masking health data to preserve confidentiality. *Statistics in medicine*, *18*, 525. https://doi.org/10.1002/(SICI)1097-0258(19990315)18:5%3C497::AID-SIM45%3E3.0.CO;2-%23

Ataei, M., Degbelo, A., Kray, C., & Santos, V. (2018). Complying with privacy legislation: From legal text to implementation of privacy-aware location-based services. *ISPRS international journal of geo-information*, *7*. https://doi.org/10.3390/ijgi7110442

Baik, J. (2020). Data privacy against innovation or against discrimination?: The case of the California Consumer Privacy Act (CCPA). *Telematics and informatics*, *52*. https://doi.org/10.1016/j.tele.2020.101431

Bellavista, P., Kupper, A., & Helal, S. (2008). Location-Based Services: Back to the Future. *IEEE pervasive computing*, *7*, 89. https://doi.org/10.1109/MPRV.2008.34

Charleux, L., & Schofield, K. (2020). True spatial k-anonymity: Adaptive areal elimination vs. adaptive areal masking. *Cartography and geographic information science*, *47*(6), 537-549.

City of Vienna. (2024). *Statistics - Vienna in Figures*. Retrieved 12 July 2024 from https://www.wien.gv.at/english/administration/statistics/

Cremonini, M., Braghin, C., & Agostino Ardagna, C. (2013). Chapter 42 - Privacy on the Internet. In (Second Edition ed., pp. 753). https://doi.org/10.1016/B978-0-12-394397-2.00042-8

Curry, D. (2024). *Strava Revenue and Usage Statistics (2024)*. Retrieved 27 January 2024 from https://www.businessofapps.com/data/strava-statistics/

Dhondt, K., Le Pochat, V., Voulimeneas, A., Joosen, W., & Volckaert, S. (2022). A Run a Day Won't Keep the Hacker Away: Inference Attacks on Endpoint Privacy Zones in Fitness Tracking Social Networks.

Esri. (2024a). *Introduction to ArcGIS Pro*. Retrieved 12 July 2024 from https://pro.arcgis.com/en/pro-app/latest/get-started/get-started.htm

Esri. (2024b). *GPX To Features (Conversion)*. Retrieved 22 July 2024 from https://pro.arcgis.com/en/pro-app/latest/tool-reference/conversion/gpx-to-features.htm

Georgiadou, Y., De By, R. A., & Kounadi, O. (2019). Location privacy in the wake of the GDPR. *ISPRS international journal of geo-information*, *8*. https://doi.org/10.3390/ijgi8030157

GitHub. (2024). *About GitHub and Git*. Retrieved 12 July 2024 from https://docs.github.com/en/get-started/start-your-journey/about-github-and-git

Goldman, E. (2020). An introduction to the california consumer privacy act (ccpa). *Santa Clara Univ. Legal Studies Research Paper*.

Grundy, Q., Held, F. P., & Bero, L. A. (2017). Tracing the potential flow of consumer data: A network analysis of prominent health and fitness apps. *J Med Internet Res*, *19*, e233. https://doi.org/10.2196/jmir.7347

Hampton, K. H., Fitch, M. K., Allshouse, W. B., Doherty, I. A., Gesink, D. C., Leone, P. A., Serre, M. L., & Miller, W. C. (2010). Mapping Health Data: Improved Privacy Protection With Donut Method Geomasking. *Am J Epidemiol*, *172*, 1069. https://doi.org/10.1093/aje/kwq248

Hassan, W. U., Hussain, S., & Bates, A. (2018). Analysis of privacy protections in fitness tracking social networks-or-you can run, but can you hide? 27th USENIX Security Symposium (USENIX Security 18),

Hinds, J., Williams, E. J., & Joinson, A. N. (2020). "It wouldn't happen to me": Privacy concerns and perspectives following the Cambridge Analytica scandal. *International journal of human-computer studies*, *143*. https://doi.org/10.1016/j.ijhcs.2020.102498

Huang, H., Gartner, G., Krisp, J. M., Raubal, M., & Van de Weghe, N. (2018). Location based services: ongoing evolution and research agenda. *Journal of location based services*, *12*, 93. https://doi.org/10.1080/17489725.2018.1508763

Kao, C.-H., Hsieh, C.-H., Chu, Y.-F., Kuang, Y.-T., & Yang, C.-K. (2017). Using data visualization technique to detect sensitive information re-identification problem of real open dataset. *Journal of systems architecture*, *80*, 91. https://doi.org/10.1016/j.sysarc.2017.09.009

Karney, C. F. F. (2013). Algorithms for geodesics. *Journal of geodesy*, *87*, 55. https://doi.org/10.1007/s00190-012-0578-z

Keßler, C., & McKenzie, G. (2018). A geoprivacy manifesto. *Transactions in GIS*, *22*, 19. https://doi.org/10.1111/tgis.12305

Kounadi, O., & Leitner, M. (2014). Why Does Geoprivacy Matter? The Scientific Publication of Confidential Data Presented on Maps. *J Empir Res Hum Res Ethics*, *9*, 45. https://doi.org/10.1177/1556264614544103

Kounadi, O., & Leitner, M. (2016). Adaptive areal elimination (AAE): A transparent way of disclosing protected spatial datasets. *Computers, environment and urban systems*, *57*, 67. https://doi.org/10.1016/j.compenvurbsys.2016.01.004

Leitner, M., & Curtis, A. (2004). Cartographic Guidelines for Geographically Masking the Locations of Confidential Point Data. *Cartographic perspectives*, 39. https://doi.org/10.14714/CP49.439

Li, J. (2015). A privacy preservation model for health-related social networking sites. *J Med Internet Res*, *17*, e168. https://doi.org/10.2196/jmir.3973

Liu, B., Zhou, W., Zhu, T., Gao, L., & Xiang, Y. (2018). Location Privacy and Its Applications: A Systematic Study. *IEEE access*, *6*, 17624. https://doi.org/10.1109/ACCESS.2018.2822260

Liz, S. (2018). U.S. soldiers are revealing sensitive and dangerous information by jogging. *The Washington Post*. https://www.washingtonpost.com/world/a-map-showing-the-users-of-fitness-devices-lets-the-world-see-where-us-soldiers-are-and-what-they-are-doing/2018/01/28/86915662-0441-11e8-aa61-f3391373867e_story.html

Martin, K., & Nissenbaum, H. (2020). What Is It About Location? *Berkeley technology law journal*, *35*. https://doi.org/10.15779/Z382F7JR6F

McKenzie, G., Romm, D., Zhang, H., & Brunila, M. (2022). PrivyTo: A privacy-preserving location-sharing platform. *Transactions in GIS*, *26*(4), 1703-1717.

Meg. (2023). *Edit Map Visibility*. Strava. Retrieved 27 January 2024 from https://support.strava.com/hc/en-us/articles/115000173384

Mink, J., Yuile, A. R., Pal, U., Aviv, A. J., & Bates, A. (2022). Users Can Deduce Sensitive Locations Protected by Privacy Zones on Fitness Tracking Apps. Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems,

Openrouteservice. (2024). Retrieved 11 July 2024 from https://openrouteservice.org/

OpenStreetMap. (2024). *Overpass API*. OpenStreetMap Wiki. Retrieved 17 July 2024 from https://wiki.openstreetmap.org/wiki/Overpass_API

Patrick, K. M. D. M. S., Griswold, W. G. P., Raab, F., & Intille, S. S. P. (2008). Health and the Mobile Phone. *Am J Prev Med*, *35*, 181. https://doi.org/10.1016/j.amepre.2008.05.001

Polzin, F., & Kounadi, O. (2021). Adaptive voronoi masking: A method to protect confidential discrete spatial data. 11th International Conference on Geographic Information Science (GIScience 2021)-Part II,

Raper, J., Gartner, G., Karimi, H., & Rizos, C. (2007). Applications of location-based services: a selected review. *Journal of location based services*, *1*, 111. https://doi.org/10.1080/17489720701862184

Richter, W. (2018). The verified neighbor approach to geoprivacy: An improved method for geographic masking. *J Expo Sci Environ Epidemiol*, *28*, 118. https://doi.org/10.1038/jes.2017.17

Samarati, P., & Sweeney, L. (1998). Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression.

Seidl, D. E., Jankowski, P., Clarke, K. C., & Nara, A. (2020). Please Enter Your Home Location: Geoprivacy Attitudes and Personal Location Masking Strategies of Internet Users. *Annals of the American Association of Geographers*, *110*, 605. https://doi.org/10.1080/24694452.2019.1654843

Seidl, D. E., Paulus, G., Jankowski, P., & Regenfelder, M. (2015). Spatial obfuscation methods for privacy protection of household-level data. *Applied geography (Sevenoaks)*, *63*, 263. https://doi.org/10.1016/j.apgeog.2015.07.001

Statista. (2023). *Number of smartphones sold to end users worldwide from 2007 to 2022*. Statista. Retrieved 6 December 2023 from https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007/

Stopher, P., FitzGerald, C., & Zhang, J. (2006). Advances in GPS Technology for Measuring Travel.

Swanlund, D., Schuurman, N., & Brussoni, M. (2020a). MaskMy.XYZ: An easy-to-use tool for protecting geoprivacy using geographic masks. *Transactions in GIS*, *24*, 401. https://doi.org/10.1111/tgis.12606

Swanlund, D., Schuurman, N., Zandbergen, P., & Brussoni, M. (2020b). Street masking: A network-based geographic mask for easily protecting geoprivacy. *Int J Health Geogr*, *19*, 26. https://doi.org/10.1186/s12942-020-00219-z

Sweeney, L. (2000). Simple demographics often identify people uniquely. *Health (San Francisco)*, *671*(2000), 1-34.

ur Rehman, I. (2019). Facebook-Cambridge Analytica data harvesting: What you need to know. *Library philosophy and practice*, *2019*, 11.

Wang, J., & Kwan, M. P. (2020). Daily activity locations k-anonymity for the evaluation of disclosure risk of individual GPS datasets. *Int J Health Geogr*, *19*, 7. https://doi.org/10.1186/s12942-020-00201-9

Westin, A. F. (1966). Science, Privacy, and Freedom: Issues and Proposals for the 1970's. Part I--The Current Impact of Surveillance on Privacy. *Columbia law review*, *66*, 1050. https://doi.org/10.2307/1120997

Xie, S. (2021). The Program Construction Method of Navigation Format Files GPX and KML Based on Geological Exploration Point Information. *Agricultural biotechnology (Pawtucket, R.I.)*, *10*, 86.

Zhang, H., & McKenzie, G. (2023). Rehumanize geoprivacy: from disclosure control to human perception. *GeoJournal*, *88*, 208. https://doi.org/10.1007/s10708-022-10598-4

Zhang, S., Freundschuh, S. M., Lenzer, K., & Zandbergen, P. A. (2017). The location swapping method for geomasking. *Cartography and geographic information science*, *44*, 34. https://doi.org/10.1080/15230406.2015.1095655

Appendix A – Code for removing BOM in a GPX file

```
def remove_bom_encoding(file_path):
    with open(file_path, 'rb') as f:
        gpx_content = f.read()
    # If BOM is present, remove it
    if gpx_content.startswith(b'\xef\xbb\xbf'):
        gpx_content = gpx_content[3:]
    with open(file_path, 'wb') as f:
        f.write(gpx_content)

file_path = "ENTER PATH TO GPX FILE HERE"
remove_bom_encoding(file_path)
```

Appendix B – Code for ska-based Privacy Method

```
# coding: utf-8
# -----------------
# created: 20.05.2024
# author: Robby Heusequin
# purpose: ska-based privacy protection method developed for master thesis at the University of Vienna
# ----------------------------------------------

# Import the necessary libraries - note might have to install the libraries
import gpxpy
import requests
import random
import openrouteservice
import os
from geopy.distance import geodesic
from geopy.geocoders import Nominatim
from shapely.geometry import Point, MultiPoint
from shapely.ops import nearest_points


'''
FUNCTIONS
'''


# Function to import gpx file, modify (delete first and last x points) and save it
def modify_gpx_file(input_gpx_file, output_gpx_file):
    # Load the GPX file
    with open(input_gpx_file, 'r') as gpx_file:
        gpx = gpxpy.parse(gpx_file)

    # Access the segment in the gpx file
    segment = gpx.tracks[0].segments[0]

    # Extract the original start and end locations
    original_start_location = (segment.points[0].latitude, segment.points[0].longitude)
    original_end_location = (segment.points[-1].latitude, segment.points[-1].longitude)

    # Print the original start and end locations
    print(f"Original Start Location: {original_start_location}")
    print(f"Original End Location: {original_end_location}")

    # Remove the first 5 and last 5 points
    segment.points = segment.points[5:-5]

    # Extract the new start and end locations
    new_start_location = (segment.points[0].latitude, segment.points[0].longitude)
    new_end_location = (segment.points[-1].latitude, segment.points[-1].longitude)

    # Print the new start and end locations
    print(f"New Start Location: {new_start_location}")
    print(f"New End Location: {new_end_location}")
```

```
52
53      # Save the modified GPX file
54      with open(output_gpx_file, 'w') as output_gpx_file:
55          output_gpx_file.write(gpx.to_xml())
56
57      # Return the original and new locations to use in the algorithm
58      locations = {
59          'original_start_location': original_start_location,
60          'original_end_location': original_end_location,
61          'new_start_location': new_start_location,
62          'new_end_location': new_end_location
63      }
64
65      return locations
66
67  # Function to import a gpx file
68  def import_gpx_file(file_path):
69      try:
70          with open(file_path, 'r') as gpx_file:
71              gpx = gpxpy.parse(gpx_file)
72              return gpx
73      except FileNotFoundError:
74          print("File not found.")
75
76
77  # Function to get the latitude and longitude from an address
78  def get_lat_lon_from_address(address):
79      geolocator = Nominatim(user_agent="ENTER EMAIL ADDRESS HERE")
80      lat_lon_address = geolocator.geocode(address)
81
82      # Return the coordinates
83      return lat_lon_address.latitude, lat_lon_address.longitude
84
85  # Get the nearest neighbour addresses from an input point and get the k
    nearest addresses
86  def k_nearest_neighbour_addresses(lat, lon, num_addresses):
87      # Create the request to ask all addresses within 1 km
88      overpass_url = http://overpass-api.de/api/interpreter
89      overpass_query =
    f"[out:json];(node[\"addr:housenumber\"](around:1000,{lat},(Liu et
    al.)););out body;"
90      overpass_response = requests.get(overpass_url, params={'data':
    overpass_query})
91      addresses_1000m = overpass_response.json()
92
93      if 'elements' in addresses_1000m:
94          addresses = []
95          for element in addresses_1000m['elements']:
96              if 'tags' in element and 'addr:housenumber' in
    element['tags']:
97                  address = element['tags'].get('addr:street', '') + ' ' +
    element['tags'].get('addr:housenumber', '') + ', ' +
    element['tags'].get('addr:postcode', '') + ', ' + str(element['lat']) +
    ', ' + str(element['lon'])
98                  address_lat = float(element['lat'])
99                  address_lon = float(element['lon'])
100                 distance = geodesic((lat, lon), (address_lat,
    address_lon)).kilometers
101
102                 # exclude the original address out of the list with a
    tolerance of 0,0001 (sometimes the same address is multiple times in the
    dataset with a slightly different lat and lon)
```

```
103                 tolerance = 0.0001
104                 if not (abs(address_lat - lat) <= tolerance and
    abs(address_lon - lon) <= tolerance):
105                     addresses.append((address, distance))
106
107                 # Create an empty set to store unique addresses - do
    this because a set does not allow duplicates
108                 addresses_found = set()
109                 addresses_no_duplicates = []
110
111                 for address, _ in addresses:
112                     # Split the address string by comma and take the
    first part
113                     address_street_number =
    address.split(',')[0].strip()
114                     if address_street_number not in addresses_found:
115                         addresses_no_duplicates.append((address, _))
116                         addresses_found.add(address_street_number)
117
118         # Sort addresses based on distance from the given lat and lon
119         # Sorts the list based on the second element of each tuple
120         addresses_no_duplicates.sort(key=lambda x: x[1])
121
122         # Return the nearest k addresses
123         return [address[0] for address in
    addresses_no_duplicates[:num_addresses]]
124     else:
125         return []
126
127 # Function to calculate the bounding box as this is needed for the query
    to get the closest street intersections
128 def calculate_bbox(gpx_file):
129     # Read the GPX file
130     with open(gpx_file, 'r') as f:
131         gpx = gpxpy.parse(f)
132
133     # Initialize variables to store min and max latitudes and longitudes
134     min_lat, max_lat = float('inf'), float('-inf')
135     min_lon, max_lon = float('inf'), float('-inf')
136
137     # Iterate over track points
138     for track in gpx.tracks:
139         for segment in track.segments:
140             for point in segment.points:
141                 # Update min and max latitudes and longitudes
142                 min_lat = min(min_lat, point.latitude)
143                 max_lat = max(max_lat, point.latitude)
144                 min_lon = min(min_lon, point.longitude)
145                 max_lon = max(max_lon, point.longitude)
146
147     # Add buffer to the bounding box
148     buffer = 0.01
149     min_lat -= buffer
150     max_lat += buffer
151     min_lon -= buffer
152     max_lon += buffer
153
154     # Define the bounding box (bbox)
155     bbox = (min_lat, min_lon, max_lat, max_lon)
156     return bbox
157
158 # Function to get the closest street intersections
```

```
def get_closest_intersections(bbox):
    overpass_url = http://overpass-api.de/api/interpreter

    # Overpass Turbo overpass_query template
    overpass_query = f"""
   [out:json][timeout:60][bbox:{bbox[0]},{bbox[1]},{bbox[2]},{bbox[3]}];

way["highway"~"^(trunk|primary|secondary|tertiary|unclassified|residenti
al|pedestrian|living_street)$"]->.streets;
    node(way_link.streets:3-)->.connections;
    foreach .connections->.connection(
      way(bn.connection);
      if (u(t["name"]) == "< multiple values found >") {{
        (.connection;.intersections;)->.intersections;
      }}
    );
    .intersections out geom;
    """

    overpass_response = requests.post(overpass_url, data={'data':
overpass_query})

    # Check if the request was successful
    if overpass_response.status_code == 200:
        # Parse the JSON overpass_response
        intersection_data = overpass_response.json()

        # Extract intersections
        intersections = [
            {'lat': element['lat'], 'lon': element['lon']}
            for element in intersection_data['elements'] if
element['type'] == 'node'
        ]

        return intersections
    else:
        print(f"Error: {overpass_response.status_code}")
        print(overpass_response.text)
        return None

# Create a gpx track
def create_gpx_segment(route_coordinates):
    gpx_segment = gpxpy.gpx.GPXTrackSegment()
    for coordinates in route_coordinates:
        lon, lat = coordinates  # OpenRouteService returns coordinates
in [lon, lat] format
        gpx_segment.points.append(gpxpy.gpx.GPXTrackPoint(lat, lon))
    return gpx_segment

# Function to calculate the total distance of a track in a gpx file
def calculate_total_distance(file_path):
    # Parse the GPX file
    with open(file_path, 'r') as gpx_file:
        gpx = gpxpy.parse(gpx_file)

    total_distance = 0.0
    for track in gpx.tracks:
        for segment in track.segments:
            total_distance += segment.length_3d()

    return total_distance / 1000.0  # Convert meters to kilometers

```

```
216
217 '''
218 DEFINE VARIABLES
219 '''
220
221
222 # Define the path where the gpx file is saved and a modified version
    will be saved
223 input_gpx_file = "ENTER FILE PATH HERE"
224 output_gpx_file = "ENTER FILE PATH HERE"
225
226 # API key for the OpenRouteServices client to request a route
227 client = openrouteservice.Client(key=ENTER OPENROUTESERVICES API KEY
    HERE)
228
229 # Import the gpx file and extract the original and temporary start and
    end point coordinates from the input gpx file
230 gpx_data_info = modify_gpx_file(input_gpx_file, output_gpx_file)
231
232 # Save the coordinates of the original and temporary start and end
    locations in a variable
233 original_start = gpx_data_info['original_start_location']
234 original_end = gpx_data_info['original_end_location']
235 temporary_start = gpx_data_info['new_start_location']
236 temporary_end = gpx_data_info['new_end_location']
237
238 # Import the modified gpx file with the new temporary start and end
    locations
239 gpx_data = import_gpx_file(output_gpx_file)
240
241 # Define the k-min and k-max for the nearest addresses
242 kMin_num_addresses = 5
243 kMax_num_addresses = 20
244
245
246 '''
247 START LOCATION
248 '''
249
250
251 # Get the k-min nearest addresses from the starting location
252 kMin_nearest_addresses_start =
    k_nearest_neighbour_addresses(original_start[0], original_start[1],
    kMin_num_addresses)
253
254 # Shifted address k-min from starting address
255 if kMin_nearest_addresses_start:
256     print(f"Nearest {kMin_num_addresses} addresses:")
257     for kMin_address_start in kMin_nearest_addresses_start:
258         print(kMin_address_start)
259 else:
260     print("No addresses found within the specified range.")
261
262 # Select a random address from the list -> k-min shifted address
263 kMin_shifted_start_address = random.choice(kMin_nearest_addresses_start)
264 print(f"the shifted start address (k-min) is:
    {kMin_shifted_start_address}.")
265
266 # Save the coordinates from the k-min shifted start address
267 kMin_shifted_address_start_coordinates =
    get_lat_lon_from_address(kMin_shifted_start_address)
268
```

```
269   # Get the k-max nearest addresses from the shifted k-min start location
270   kMax_nearest_addresses_start =
      k_nearest_neighbour_addresses(kMin_shifted_address_start_coordinates[0],
      kMin_shifted_address_start_coordinates[1], kMax_num_addresses)
271
272   # Shifted address k-max from the shifted k-min address
273   if kMax_nearest_addresses_start:
274       print(f"Nearest {kMax_num_addresses} addresses:")
275       for kMax_address_start in kMax_nearest_addresses_start:
276           print(kMax_address_start)
277   else:
278       print("No addresses found within the specified range.")
279
280   # Select a random address from the list -> k-max shifted address
281   kMax_shifted_address_start = random.choice(kMax_nearest_addresses_start)
282   print(f"the shifted start address (k-max) is:
      {kMax_shifted_address_start}.")
283
284   # Save the coordinates from the k-max shifted start address
285   kMax_shifted_start_address_coordinates =
      get_lat_lon_from_address(kMax_shifted_address_start)
286
287   # Get the bounding box for querying the street intersections
288   # bbox (south, west, north, east)
289   bbox = calculate_bbox(input_gpx_file)
290   print(f"the bounding box is: {bbox}.")
291
292   # Find the nearest intersection point
293   intersections_start = get_closest_intersections(bbox)
294
295   # Create a Shapely Point object for the k-max shifted start address
296   original_point_start = Point(kMax_shifted_start_address_coordinates[0],
      kMax_shifted_start_address_coordinates[1])
297   print(original_point_start)
298
299   # Create a list to store Shapely Point objects for the intersections
      found inside the bounding box
300   intersection_points = [Point(intersection['lat'], intersection['lon'])
      for intersection in intersections_start]
301
302   # Create a MultiPoint object to store all the intersections
303   multi_points = MultiPoint(intersection_points)
304
305   # Find the nearest intersection point
306   nearest_intersection_start = nearest_points(original_point_start,
      multi_points)[1]
307
308   # Extract the latitude and longitude of the closest street intersection
309   nearest_intersection_start_lat = nearest_intersection_start.x
310   nearest_intersection_start_lon = nearest_intersection_start.y
311
312   print(f"The nearest intersection is at {nearest_intersection_start_lat},
      {nearest_intersection_start_lon}")
313
314   # Save the coordinates in a variable
315   nearest_intersection_start_lat_lon = nearest_intersection_start_lat,
      nearest_intersection_start_lon
316
317   # Get the coordinates of the closest street intersection and the
      temporary start location
318   start_coordinates_lat_lon = ((nearest_intersection_start_lat_lon),
      (temporary_start))
```

```
319  print(start_coordinates_lat_lon)
320  # Convert to (longitude, latitude) format - OpenRouteServices needs the
     coordinates in lon, lat!
321  start_coordinates_lon_lat = [(lon, lat) for lat, lon in
     start_coordinates_lat_lon]
322  print(start_coordinates_lon_lat)
323
324  # Request the new walking route from the OpenRouteServices API
325  shifted_start_route = client.directions(
326      coordinates=start_coordinates_lon_lat,
327      profile='foot-walking',
328      format='geojson'
329  )
330
331  # Extract the coordinates of the start route
332  start_route_geometry = shifted_start_route['features'][0]['geometry']
333  start_route_coordinates = start_route_geometry['coordinates']
334
335  # Create a new GPX track from the route coordinates for the start
     location
336  start_gpx_segment = create_gpx_segment(start_route_coordinates)
337
338  # Insert the new segment into the GPX track at the first position
339  gpx_data.tracks[0].segments.insert(0, start_gpx_segment)
340
341
342  '''
343  END LOCATION
344  '''
345
346
347  # Get the k-min nearest addresses from the end location
348  kMin_nearest_addresses_end =
     k_nearest_neighbour_addresses(original_end[0], original_end[1],
     kMin_num_addresses)
349
350  # Shifted address k-min from end address
351  if kMin_nearest_addresses_end:
352      print(f"Nearest {kMin_num_addresses} addresses:")
353      for kMin_address_end in kMin_nearest_addresses_end:
354          print(kMin_address_end)
355  else:
356      print("No addresses found within the specified range.")
357
358  # Select a random address from the list -> k-min shifted address
359  kMin_shifted_address_end = random.choice(kMin_nearest_addresses_end)
360  print(f"the shifted end address (kMin) is: {kMin_shifted_address_end}.")
361
362  # Save the coordinates from the k-min shifted end address
363  kMin_shifted_address_end_coordinates =
     get_lat_lon_from_address(kMin_shifted_address_end)
364
365  # Get the k-max nearest addresses from the shifted k-min end location
366  kMax_nearest_addresses_end =
     k_nearest_neighbour_addresses(kMin_shifted_address_end_coordinates[0],
     kMin_shifted_address_end_coordinates[1], kMax_num_addresses)
367
368  # Shifted address k-max from the shifted k-min address
369  if kMax_nearest_addresses_end:
370      print(f"Nearest {kMax_num_addresses} addresses:")
371      for kMax_address_end in kMax_nearest_addresses_end:
372          print(kMax_address_end)
```

```
373   else:
374       print("No addresses found within the specified range.")
375
376   # Select a random address from the list -> k-max shifted address
377   kMax_shifted_address_end = random.choice(kMax_nearest_addresses_end)
378   print(f"the shifted end address (kMax) is: {kMax_shifted_address_end}.")
379
380   # Save the coordinates from the k-max shifted end address
381   kMax_shifted_address_end_coordinates =
      get_lat_lon_from_address(kMax_shifted_address_end)
382
383   # Find the nearest intersection point
384   intersections_end = get_closest_intersections(bbox)
385
386   # Create a Shapely Point object for the k-max shifted end address
387   kMax_shifted_point_end = Point(kMax_shifted_address_end_coordinates[0],
      kMax_shifted_address_end_coordinates[1])
388
389   # Find the nearest intersection
390   nearest_intersection_end = nearest_points(kMax_shifted_point_end,
      multi_points)[1]
391
392   # Extract the latitude and longitude of the closest street intersection
393   nearest_intersection_end_lat = nearest_intersection_end.x
394   nearest_intersection_end_lon = nearest_intersection_end.y
395
396   print(f"The nearest intersection is at {nearest_intersection_end_lat},
      {nearest_intersection_end_lon}")
397
398   # Save the coordinates in a variable
399   nearest_intersection_end_lat_lon = nearest_intersection_end_lat,
      nearest_intersection_end_lon
400
401   # Get the coordinates of the temporary end location and the closest
      street intersection
402   end_coordinates_lat_lon = ((temporary_end),
      (nearest_intersection_end_lat_lon))
403   print(end_coordinates_lat_lon)
404   # Convert to (longitude, latitude) format - OpenRouteServices needs the
      coordinates in lon, lat!
405   end_coordinates_lon_lat = [(lon, lat) for lat, lon in
      end_coordinates_lat_lon]
406   print(end_coordinates_lon_lat)
407
408   # Request the new walking route from the OpenRouteServices API
409   shifted_end_route = client.directions(
410       coordinates=end_coordinates_lon_lat,
411       profile='foot-walking',
412       format='geojson'
413   )
414
415   # Extract the coordinates of the end route
416   end_route_geometry = shifted_end_route['features'][0]['geometry']
417   end_route_coordinates = end_route_geometry['coordinates']
418
419   # Create a new GPX track from the route coordinates for the end location
420   end_gpx_segment = create_gpx_segment(end_route_coordinates)
421
422   # Insert the new segment into the GPX track at the second position
423   gpx_data.tracks[0].segments.insert(2, end_gpx_segment)
424
425
```

```
426 '''
427 SAVING THE MODIFIED GPX FILE
428 '''
429
430
431 # Define the base path and the masked GPX file name
432 dir_name = "ENTER DIRECTORY HERE"
433 masked_name = "ENTER A MASKED GPX FILE NAME HERE"
434 masked_gpx_file = os.path.join(dir_name + masked_name)
435
436 # Save the new GPX file
437 with open(masked_gpx_file, 'w') as f:
438     f.write(gpx_data.to_xml())
439
440 print(f"GPX file has been saved as {masked_name}")
441
442 # Calculate the original and masked total distance travelled in the GPX
    files
443 total_distance_original_gpx = calculate_total_distance(input_gpx_file)
444 total_distance_masked_gpx = calculate_total_distance(masked_gpx_file)
445
446 print(f"The total distance travelled of the original gpx file:
    {total_distance_original_gpx:.2f} km.")
447 print(f"The total distance travelled of the masked gpx file:
    {total_distance_masked_gpx:.2f} km.")
```