Location spoofing attack detection with pre-installed sensors in mobile devices

Shing Ki Wong and Siu Ming Yiu*
The University of Hong Kong, Hong Kong
{skwong, smyiu}@cs.hku.hk

Received: September 30, 2020; Accepted: November 28, 2020; Published: December 31, 2020

Abstract

The Global Positioning System (GPS) plays an important role in many industries nowadays. It provides a simple and convenient way for users to locate their position on the Earth. Since the radio wave signal from the GPS Satellite propagates through space and air, it can be easily altered by location spoofing attack (LSA), which is a signal interference on legitimate GPS signal in order to transmit inaccurate GPS coordinates to the target. In recent years, the term 'location spoofing' can also refer to the act of false reporting on one's GPS location to other location-based applications. Not many research works have been proposed against the latter problem and none of them provide a good solution on detecting location spoofing within local areas. In this paper, we clarify and differentiate between these two kinds of location spoofing attacks to avoid ambiguity in future research work. We proposed a behavioral detection method making use of the gyroscope and accelerometer commonly equipped in most mobile devices. We verify the genuineness of the GPS data by comparing the travel direction with the facing direction deduced from the orientation data provided by the gyroscope. We further examine the GPS data with the step count provided by the accelerometer to see whether the step length of the travel aligns with the average adult step length. Experiment results show that our proposed methodology can efficiently differentiate between normal and spoofing travels with large deviation on travel direction and step length. We also show that our method is simple to implement in practical situations.

Keywords: Location Spoofing, GPS Spoofing, Gyroscope, Accelerometer, Mobile Sensor

1 Introduction

The Global Positioning System (GPS) has been developed of a few decades. It is originally designed for military and aviation purposes while it is commonly used by many other services nowadays. Under the line of sights of four or more GPS satellites, one can easily pinpoint his/her location on the Earth from the geolocation data provided by the Global Navigation Satellite System (GNSS). The contribution of this location service is critical for military, aviation and marine industries, providing massive convenience to human beings. With the rapid growth on the development of the Global Positioning System, it has now been frequently used everywhere in our daily lives. Every single smartphone nowadays has its GPS module installed to provide necessary data for location-based applications such as map navigation, public transport scheduling and even mobile games. During the past few years, several famous location-based augmented reality mobile games (e.g. Pokémon GO [7], Jurassic World Alive [3] and Harry Potter: Wizards Unite [2]) are being developed, providing an brand-new gaming experience for players. By

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 11(4):16-30, Dec. 2020
DOI:*10.22667/JOWUA.2020.12.31.016*

*Corresponding author: Department of of Computer Sciences, The University of Hong Kong, Pokfulam, Hong Kong, Tel: +852-2857-8242

making use of the GPS data from the mobile device, they integrate the real world with the virtual world. Players get the feelings of having virtual creatures (i.e. Pokémon, dinosaurs and magical creatures) appearing in their real neighborhood. They can travel from place to place in the real world to discover and interact with the virtual creatures in game. Such gaming experience is extraordinary compared to traditional mobile games and the games are showing great success with huge popularity all over the world.

Despite the great fun provided by the virtual reality, travelling from place to place while playing the games can be quite tiring. Not everyone has the time and ability to play the games whenever they want. What's more, there is a significant number of people living in countries that are not supported by these location-based mobile games. In order to enjoy such novel gaming experience in their countries, they have to conduct location spoofing in their mobile devices. In a location-based mobile game, the location of the game avatar relies on the GPS position provided by the GPS module of the mobile device. By modifying the values of these location data, players can spoof themselves to anywhere on the Earth without travelling for a single centimeter in the real world. Such kind of cheating is called Location Spoofing Attack (LSA). In fact, a large number of location spoofing applications have been already developed and can be easily found in app stores or via the Internet. They offer a wide range of functionalities, such as route planning and recording, in addition to the basic feature of location modification of the GPS data. The abuse of these location spoofing applications completely destroys the fairness between cheaters and legitimate players, hurting the whole game community and finally the game itself. Even worse, developers can earn great profit by selling their spoofing applications to the huge player base. Players are willing to pay for a few dollars to have the ability to travel in game freely without paying any physical effort. As a result, the profit of the game company is sacrificed and it will eventually hinder the development of such kind of novel mobile gaming applications.

Take Pokémon GO, the most famous title in this game category, as an example. By spoofing from place to place inside the game, players can effortlessly discover rare appearing Pokémon all over the world or hatch Pokémon eggs without walking by themselves. They gain huge advantage over other legitimate players, bringing vast dissatisfaction among the general community. To alleviate such situation, Niantic [5] conducted a number of measures to tackle the problem of location spoofing. They scan for malicious applications during runtime and prohibit the usage of mock location application and rooted devices. Flagged users will get shadow-banned so that they cannot encounter rare Pokémon. They may even get permanently banned for frequent spoofing activities. In spite of this, the problem of location spoofing persists upon the fourth-year anniversary of the game. Spoofing application developers keep updating their software to bypass the loopholes of the operating system to offer consistence support on location spoofing in the game. It appears that the current measures taken from Niantic cannot entirely solve the problem.

In this paper, we improve our previous work [21] for location spoofing detection on mobile devices in local area as follows. First, we differentiate the two different types of location spoofing to avoid ambiguity for further studies in this research foundation. Second, we provide clarification and categorization on the related research work among these two areas. Last but not least, we provide a new dimension of detection criteria by introducing the usage of footstep counter service provided by the accelerometer in combination with the GPS data to determine the average step length of the mobile user.

The rest of the paper is organized as follow. Section 2 clarifies the ambiguity of the term "location spoofing" by introducing and classifying the related work on location spoofing detection to their respective categories. Section 3 reviews certain current prevention measures against location spoofing conducted by gaming companies. Section 4 describes on the common design of GPS spoofing applications currently available in the market. Section 5 explains the concept of our proposed detection algorithm in details. Section 6 presents and demonstrates the experiment results. Section 7 points out some limitation of our detection method and suggests possible future improvement. Section 8 concludes on our research

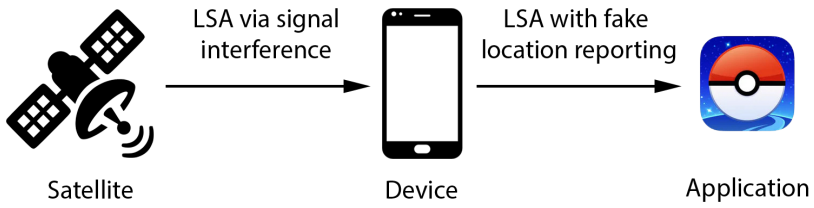


Figure 1: Location spoofing attacks in the pipeline of a GPS service.

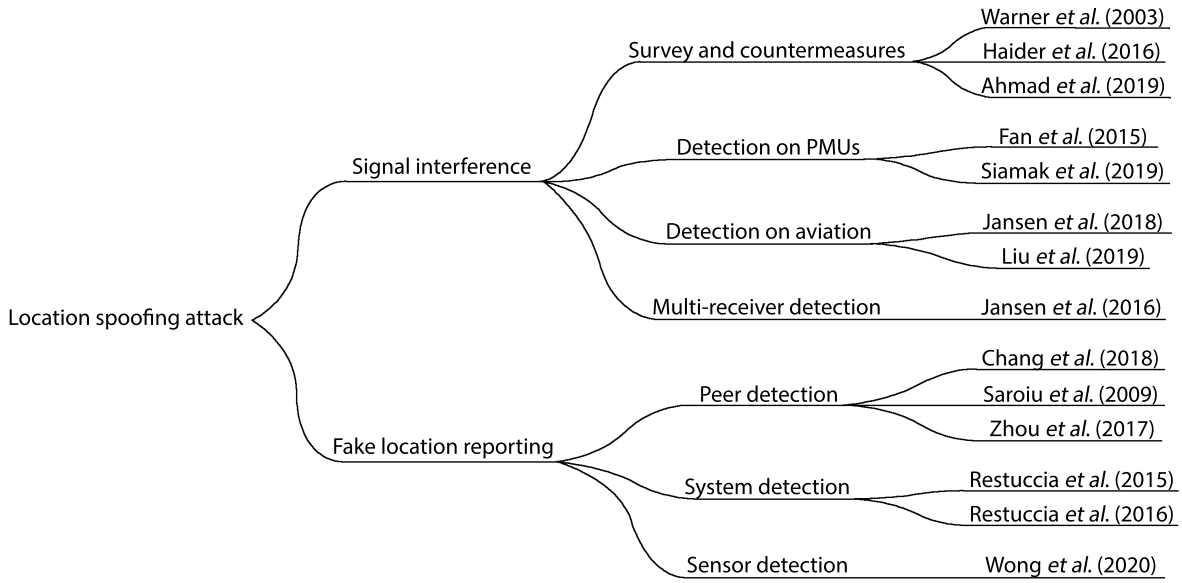


Figure 2: Literature classification and categorization with contributions from different authors.

work.

2 Related Work

2.1 Classification on location spoofing

The term location spoofing (or GPS spoofing) generally refers to an alteration on the location information. At first, it refers to an attack which a radio transmitter located near the target is used to interfere with legitimate GPS signals, providing false location information to mislead the victim. In recent years, with the wide adoption of GPS services in mobile devices, it can on the other hand refers the act of intentionally falsifying one's actual locational information. Figure 1 indicates the positions of the two location spoofing cases in the pipeline of a GPS service. The traditional location spoofing attack via signal interference occurs between the GNSS and the target device (left arrow), while the location spoofing attack on false reporting in position data occurs between the user device and the target application (right arrow). Such term is being used interchangeably in the abovementioned scenarios by researchers and gives ambiguity in the related research areas. [19] made use of a layered model to classify spoofer attacks and presented the most probable spoofing attacks with outlines of the applicable anti-spoofing methods. It classified LSA into RF based and interface based, which the former one refers to the left

hand side of the pipeline, and the latter one refers to the right-hand side of the pipeline in figure 1. It gave a comprehensive presentation on RF based spoofing attacks with counter measures, while did not mention much on interface based LSA between the user device and the target mobile application. Figure 2 classifies and categorizes the research works in these two areas.

2.2 Location spoofing via signal interference

A notable amount of research work have been done on RF based location spoofing detection. [20] acted as an entry point in this research area and proposed several countermeasures based on the signal strength difference between true and fake GPS signal. It also proposed some detection strategies by recognizing the characteristic of the satellite simulator. [11] gave a survey on effective GPS spoofing countermeasures. It overviewed the proposed techniques on protecting civilian GPS receiver and analyzed alternative solutions to prevent GPS spoofing. It also tried to explore possibilities to combine multiple prevention measures to an effective spoofing countermeasure technique. [8] pointed out the vulnerability of GPS infrastructure and provided a detailed survey on spoofing detection and countermeasure to spoofing. It presented a spoofing scenario to show that GPS spoofing detection is comparatively harder to tackle than GPD jamming. [10] proposes a cross-layer detection mechanism to detect simultaneous attacks towards multiple PMUs. It evaluated on both the physical layer information and power grid measurements to identify the PMUs being attacked. [18] proposed a GPS spoofing attack detection method in power grids using a dynamic estimator to detect the phase angle difference caused by spoofing and compared its performance with the results of other two static methods. [12] proposed a system to detect and localize GPS spoofing attacks on moving airborne targets. It monitors air traffics from GPS-derived position advertisements and detect spoofing attacks by an independent infrastructure on the ground. [14] proposed a GPS spoofing detection algorithm which analyze the received signal strength indicator (RSSI) and the timestamps at server (TSS) of ATC messages that are periodically broadcasted by aerial vehicles. [13] built the first realization of a multi-receiver GPS spoofing detection methodology making use of the reported positions by multiple GPS receivers to detect malicious spoofing signals. It gave improvement on the false acceptance rate while at the same time preserving the sensitivity to attacks.

2.3 Location spoofing with fake location reporting

Compared to location spoofing preventions on signal interference, not very much attention is being put on interface based LSA as it is a relatively new research area in location spoofing. [22] encouraged cartographers and GIScientists to pay more attention to location spoofing and suggested it as an emerging spatial data quality issue. It reviewed several frequently conducted location spoofing techniques in Pokémon GO in order to unveil the hidden motivation for location spoofing. A comprehensive case study based on Pokémon GO is given in [24] to draw scholars' attention to this emerging spoofing issue. It applies actor–network theory (ANT) to examine various human and nonhuman agents involved including gamers, spoofers, game developers, bots, and hackers. It argues that location spoofing behaviors should not be merely interpreted as fake data, while on the other hand should be considered seriously as real human geographic data.

Despite of the lack of quantity of work in this research area, several qualitied studies is being proposed in recent years and most of them make use of the network profile (e.g. IP address) in the detection process. [9] proposed a server-sided and network-based framework to detect location spoofing attack users. It tries to verify whether the location of the user's edge router's IP address matches with the GPS location provided by the user. This method is not applicable on detecting local area location spoofing which occurs only in the neighborhood within the range of the same router. It also has the constraint of using IMCP packet only. [15] proposed a location validation scheme for participatory sensing (PS)

Table 1: Comparison between our proposed scheme with existing location spoofing detection methodologies.

Proposed schemes	**Location proof**	**Secure PS system from LSAs**	**Secure LBS from LSAs**	**Fake GPS Defender**	**Sensor detection**
Author	Saroiu *et al.* (2009)	Restuccia *et al.* (2015)	Restuccia *et al.* (2016)	Chang *et al.* (2018)	Wong *et al.* (2020)
Verifier	Wi-Fi hot spots or cell towers	WiFi hot spots	WiFi hot spots	Edge routers	Mobile sensors
Evaluation	System	Peer	Peer	System	Self
Data source	Client	Client	Client	Server	Client
Architecture alteration	Yes	Yes	Yes	No	No
Deployment scale	Large	Small	Small	Large	Small
Cost	Large	Small	Small	Small	Small

systems without losing any quality of information (QoI) in the system. It assumes that users under the same Wi-Fi range of a mobile hotspot are directly connected with each other so that they can mutually validate their locations and can therefore, identify spoofing users. [16] proposed a location validation system (LVS) which verifies user locations in location-based service systems. It is based on a similar idea that devices are practically sharing the same location if they are directly connected with the same Wi-Fi. Due to the limited range of Wi-Fi signal, devices under the same Wi-Fi network can mutually validate each other's location. After multiple rounds of reputation updates, malicious users would then be filtered out. This system works well under Wi-Fi connection, while lacks practicability in cellular network due to its much larger signal range. [17] came up with location proofs which are signed by an access point and issued to devices connecting to it. It acts as a digital certificate which can be verified later by access points to provide proofs on the devices' current and previous locations. This approach is effective theoretically but it requires implementation of the module in every access point, which is impractical in a large scale. [23] proposed quantitative and qualitative approaches for location spoofing detection. It made use of millions of geo-tagged tweets to examine its proposed Bayesian time geographic detection approach. Such approach is capable to detect certain types of spoofed locations while it lacks scalability in detection on large samples.

From our observation, all the above approaches make use of the location details provided by the Wi-Fi network for authentication. However, these methodologies may not be applicable in detecting location spoofing in local area, which the user spoofs under the signal range of the same router. In this circumstance, the device will always share the same location as the router does, and no location spoofing can therefore be successfully detected. Because of this, these approaches are not practical for location-based mobile games. To tackle the problem in such situation, we proposed a novel behavioral detection methodology making use of the gyroscope equipped in mobile devices to check if the device's movement aligns with its GPS coordinates. We showed that our methodology can efficiently identify local location spoofing behaviors. We further improve our methodology in this paper by introducing an additional sensor data source provided by the accelerometer to identify location spoofing activities from a new dimension. Table 1 compares our proposed scheme with the related works.

3 Prevention measures against GPS spoofing

Currently there are several prevention measures available to combat GPS spoofing in mobile devices. They work pretty well in their targeted areas while fail in other situations. In this section, we provided a summary on these state-of-the-art common measures adopted against location spoofing.

3.1 Travel viability check

It is not a trivial task to check if a GPS location is genuine or not. However, it is possible to check whether it is actually physically reachable between any two GPS locations within a certain period of time. One can be able to determine whether a user is using location spoofing if the time elapsed between two GPS locations is too short to be physically reachable. Unfortunately, such detection is not suitable for detecting location spoofing in local area as the time elapsed are too short to be deterministic for short distance travels.

3.2 Mock location app detection

In Android OS, the mock location functionality is provided in developer options for devices without GPS modules to emulate their GPS values. It allows user devices to produce fake information on their location using the GPS and network operator. As a result, one can easily modify the GPS location via a location spoofing application by setting it as a mock location application. Because of this, the mock location application setting have to be checked and disabled via the *ALLOW_MOCK_LOCATION* attribute in the system for the sake of preventing location spoofing using this approach. However, by converting the mock location application into system application, cheaters can easily escape from such detection.

3.3 Malicious app detection

During execution, the game application tries to scan for malicious application installed in the device and stops running upon successful detection. To sustainably conduct this approach, however, frequent update and maintenance is required on the list of malicious applications. With the rapid growth on the number of location spoofing applications, such method can be quite tedious. What's more, spoofing applications can further hide themselves from detection by replacing their package names with random strings.

3.4 Emulator detection

Recently, a wide range of mobile device emulators such as Bluestack [1] and Nox App Player [6] which offer a wide range of functionalities for gaming are developed. They provide location modification and device rooting for the sake of location spoofing. Obviously, playing location-based games via theses emulators would violate the terms and condition of the games, but it is no easy to check whether the operating system is running in real devices or via emulators. By looking at the *FINGERPRINT* value of the system, one can possibly make the determination. However, such value can be easily modified by carrying out privilege escalation.

3.5 Jailbreak and root detection

By jailbreaking in IOS or rooting in Android to gaining privilege in the operating systems, users can easily modify their GPS locations afterwards. Therefore, prohibiting jailbroken or rooted devices from running the game applications becomes one of the solutions. Unfortunately, there exists third party applications that can help to hide the device from such detection (e.g. Magisk Manager [4]). They

Figure 3: Location spoofing in Pokémon GO.

can even hide other malicious applications from being detected by the game application. Such kind of applications are well supported and developed with guaranteed performance, making the location spoofing process much easier.

4 Location spoofing application

In this section, we talk about the underlying mechanism of common location spoofing applications. Generally speaking, there are five variables, including *latitude*, *longitude*, *altitude*, *speed* and *accuracy*, that a location spoofing application can modify from the location data. For spoofing applications to perform location spoofing, it is sufficient by only modifying the latitude and longitude. For simplicity, therefore, most location spoofing applications usually update the values of latitude and longitude only when they are simulating the device movement. Figure 3 shows an example on a location spoofing application (Fake GPS Location - GPS Joystick) running on top of Pokémon GO. Most of these GPS spoofing applications provide a joystick control in the user interface as shown on the right-hand side of the figure. By dragging on the joystick, the GPS coordinates will be updated correspondingly and the avatar in game will move accordingly. As a result, cheaters can simulate their desired movement in game with the spoofing application and travel from place to place effortlessly.

In practical situation, when a device refreshes its GPS location, beside the values of latitude and longitude, it's current altitude, speed and the accuracy (in meters) of how accurate the device believes those values are will also be updated. All of these five attributes fluctuate from time to time even when the device is not moving. Therefore, one can possibly tell whether a user is using location spoofing of only the latitude and longitude are being updated. Because of this, many advanced location spoofing applications will now emulate real GPS updates by modifying all of the five variables and provide customization on these values via settings. They even try to make it more "human-like" by adding random offsets to the

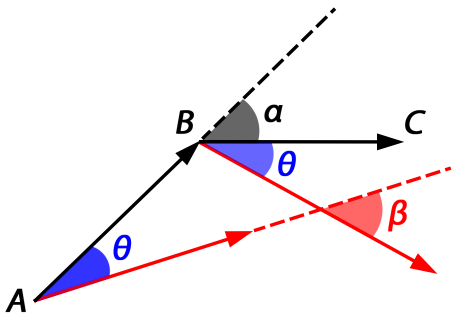


Figure 4: Illustration of the idea of our detection algorithm.

updates, making the detection work even harder.

5 Proposed method

We have learnt in section 2 it is insufficient to identify location spoofing behaviors only via system detections. In this section, we propose a behavioral detection methodology on location spoofing making use of the gyroscope and accelerometer commonly equipped in most mobile devices nowadays. We focus on scenarios which users walk around within a local area with their phones holding on their hands. We make an assumption that the back of the mobile device always faces to the same direction as the user travels, which is a common gameplay scenario for a location-based mobile game. The horizontal orientation of the mobile device with respect to the user will remain constant throughout the travel. With such a fixed horizontal orientation, we can then accurately verify whether the user's travel direction aligns with the device's facing direction.

Figure 4 illustrates our idea with an example. Suppose a user is travelling from A to B and then from B to C. The black arrows represent the travelling paths of the user moving from A to B and B to C, and the red arrows indicate the facing direction of the mobile device correspondingly. θ is the angle difference between the travelling direction of the user and facing direction of the mobile device. α is the travel direction change from AB to BC and β is the corresponding facing direction change of the mobile device. For a legitimate travel, we expect $\alpha \approx \beta$ as long as θ remains stable. If the user is using location spoofing application to manipulate the travelling path with himself/herself not moving actually, β is not going to synchronize with α since the device will always be facing to the same direction. Therefore, by looking at the difference between α and β, we can differentiate between legitimate users and spoofing users.

Apart from monitoring the angle differences, we also keep track on the step count of the device making use of the footstep counter provided by the accelerometer. Since one of the main reasons for location spoofing is to save the time on travelling from place to place physically, it is very likely that the location spoofer is stationary when he/she is spoofing. Therefore, it is expected that the total number of steps taken by the location spoofers will be significantly lower than other legitimate users, resulting in a much bigger value of average step length. By comparing this step length with the average adult step length, we can have a good indicator on whether a user is conducting location spoofing.

5.1 Methodology

Our detection method collects three sets of data. The first set of data is the GPS coordinates from the GPS module of the device. The second set of data is the orientation of the device from the gyroscope

and the third set of data is the step count from the accelerometer of the device. To determine whether the travel direction of the user aligns with the facing direction of the device, we compute the horizontal travel directions of the user from the GPS data and compare it with the facing angles of the device deduced from the orientation data. Afterwards, we calculate the average of these differences, denoted as Δ_θ, throughout the travel and compare it with the benchmark, which will be mentioned in the section 5.2, to identify potential location spoofing behaviors. To obtain the average step length of the user, denoted as len_s, we calculate the total travel distance from the GPS data and divide it by the total number of steps taken. Algorithm 1 describes the detailed procedure of our methodology.

Algorithm 1: Calculate Δ_θ and len_s from GPS coordinates, device orientation and footstep counter

$TotalDistance \leftarrow 0$;
foreach GPS_i, GPS_{i+1} **do**
  $t_1 \leftarrow GPS_i.timestamp()$;
  $t_2 \leftarrow GPS_{i+1}.timestamp()$;
  $\theta_1 \leftarrow DirectionFrom(GPS_i, GPS_{i+1})$;
  $\theta_2 \leftarrow AverageOrientation(t_1, t_2)$;
  $AngleDifference.append(AbsoluteDifference(\theta_1, \theta_2))$;
  $TotalDistance \leftarrow TotalDistance + DistanceFrom(GPS_i, GPS_{i+1})$;
end
$\Delta_\theta \leftarrow mean(AngleDifference)$;
$len_s \leftarrow TotalDistance/Steps.count()$;

5.2 Benchmark

The accuracy of our detection algorithm depends on the complexity of the path. If a user spoofs with a path that coincidently aligns with the device's facing directions throughout the travel, the detection may become unsuccessful. In this section, we set up a benchmark for Δ_θ in general spoofing situations. We assume that in the long run, the travelling path of a user will be arbitrary with equal probability in every direction. With such assumption, we can simulate a general path with 361 GPS positions and 360 random path fragments, where each path fragment is a travel with a direction within 0° to 359° with respect to the north. During spoofing, the facing direction of the mobile device should remain constant. Therefore, the average angle differences between the device's travel direction and facing direction for such a path in spoofing scenario would be

$$\Delta_\theta = \frac{0^\circ + 1^\circ + ... + 179^\circ + 180^\circ + 179^\circ + ... + 1^\circ + 0^\circ}{360} = 90^\circ$$

Claim: For a random spoofing path, the average angle differences between the device's travel direction and facing direction during spoofing is 90°.

6 Evaluation

6.1 Experiment setup

We conducted sample walks in various open areas in Hong Kong. Figure 5 to 7 show the GPS mappings of the sample walks with the mobile device travelling in (i) straight path (Fig. 5), (ii) circular path (Fig. 6) and (iii) complex paths (Fig. 7). Throughout each sample walk, the device is being held with its rear

Figure 5: Device travelling in straight path.

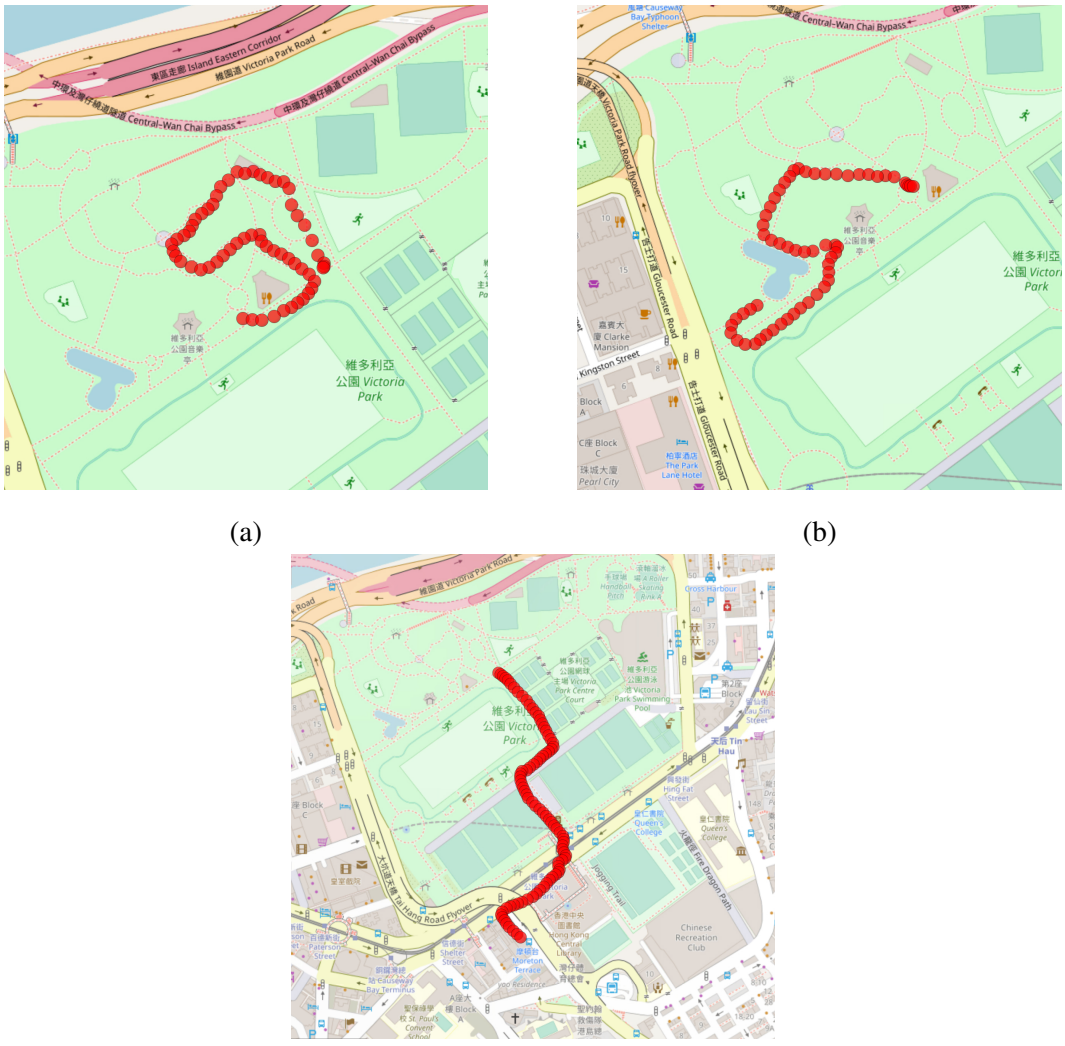
Figure 6: Device travelling in circular path.

Table 2: Average of the angle differences between the device's GPS travel direction and facing direction in different paths, with their corresponding average step lengths.

Path	Δ_θ	Δ_θ (spoofing)	No. of steps	Total distance (m)	len_s (m)
Straight	12.5105°	51.9088°	343	245.8834	0.7168
Circular	14.3276°	92.3563°	593	418.2224	0.7053
Complex (a)	17.1231°	99.8231°	526	348.2545	0.6621
Complex (b)	15.8670°	107.9607°	500	332.0762	0.6642
Complex (c)	14.2551°	141.4167°	771	518.8709	0.6730

side facing to the front. The GPS coordinates are being logged with minimal and desired update intervals of 5 and 10 seconds respectively. The facing direction of the device with respect to the north are being logged with an update interval of 500 milliseconds. Within the intervals between any two consecutive GPS coordinates, the horizontal direction of travel, the average value of the facing angles and the travel distance of the device are being calculated for subsequent computations of Δ_θ and len_s. What's more, to simulate the location spoofing scenarios of the sample walks for head to head comparison later on, we fix the orientation of the device at 0° throughout the walk to obtain the corresponding values of Δ_θ for the spoofing cases.

As we have assumed that the orientation of the device with respect to the user remains constant throughout the walks, their horizontal rotational changes should also be the same throughout the walks. Therefore, we can expect that the travel direction and the facing angle direction of the device should match with each other throughout the travels. We can also expect that the average step length of the travels should comply with the average step length of adult ($\approx$ 0.67 to 0.76 m). If there exists notable angle differences between the two directions (i.e. $\Delta_\theta \geq 90°$), or the average step length is notably larger than the average stride length of adult ($\approx$ 1.32 to 1.48 m), we can conclude that there are location spoofing behaviors identified.



(a) (b)

(c)

Figure 7: Device travelling in complex paths.

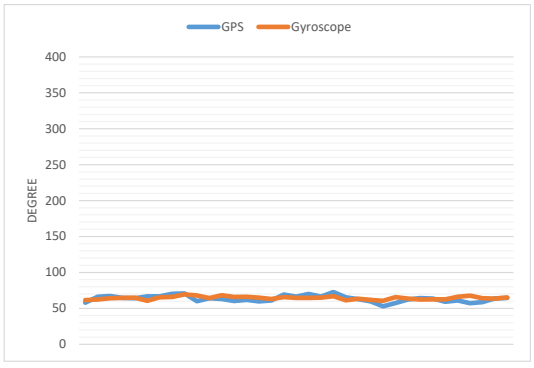


Figure 8: Bearing comparison for device travelling in straight path.

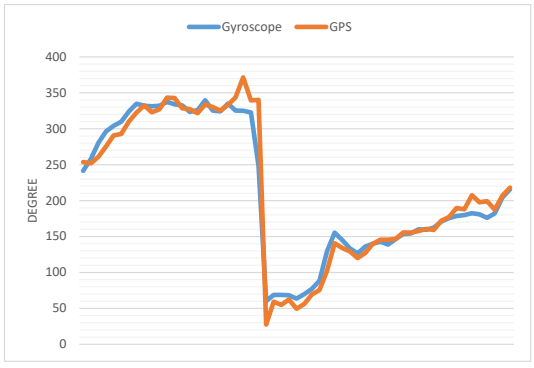
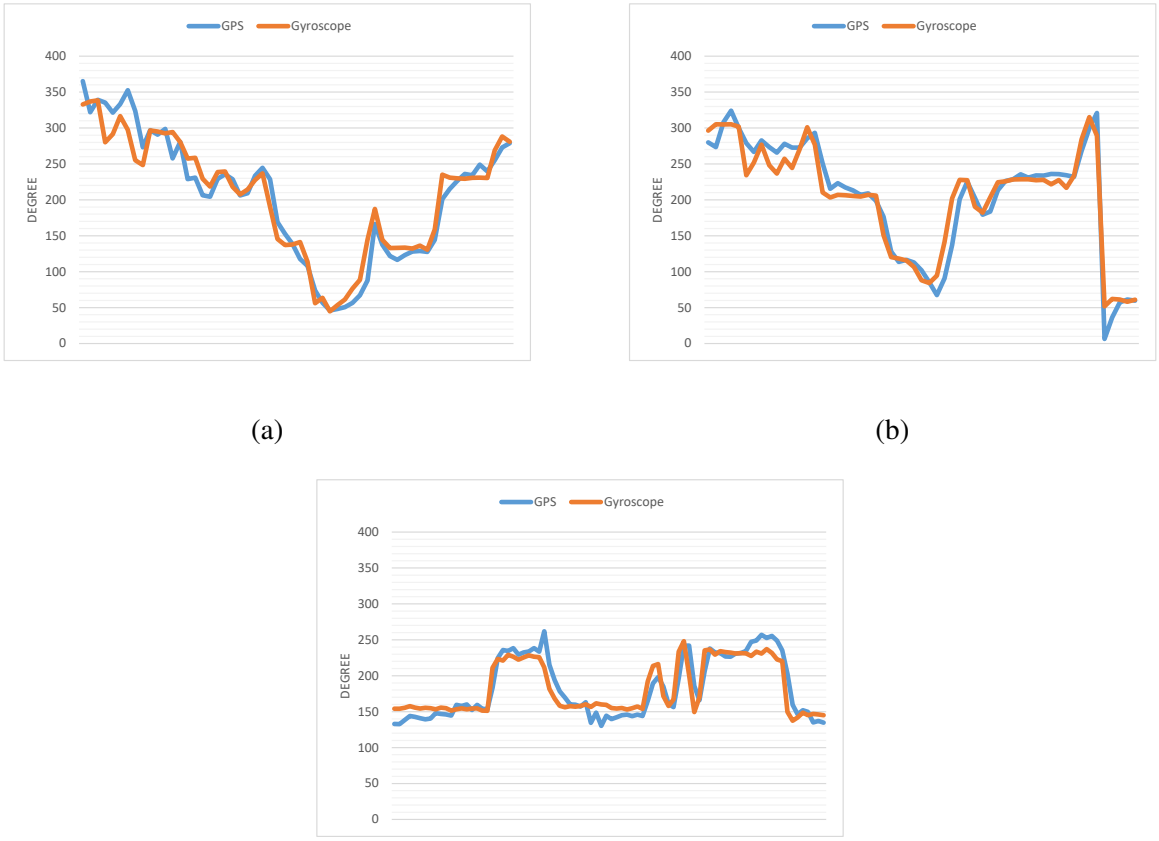


Figure 9: Bearing comparison for device travelling in circular path.



(a) (b)

(c)

Figure 10: Bearing comparison for device travelling in complex paths.

6.2 Results

Figure 8, 9 and 10 show the bearing comparison between the calculated bearings from the GPS coordinates and the corresponding facing direction of the mobile device. Since the facing direction of the mobile device will not always align perfectly with the travel direction of the user, we calculate the average dispersion between the two directions and apply a shift to align the two angles together to give a better visualization for comparison. We can see that the two bearings variate with similar trends, implying that the device's facing angle changes matches with the direction changes of its GPS path. Table 2 summarizes the average angle differences between the travel direction and facing direction of the device in our sample paths, with the corresponding total travel distances and steps taken. We can see that in spoofing cases (3rd column), Δ_θ retains at much high values ($> 90^\circ$ except for straight path) compared to that in real cases ($< 15^\circ$) (2nd column). As mentioned in section 5.2, our methodology depends on path complexity. For straight paths which only involve small direction changes, the corresponding Δ_θ for spoofing case depends heavily on the dominant moving direction of the path, which can therefore result in relatively random outcomes (51.9088° in our sample case). For circular path with even distribution in travelling directions (figure 6), Δ_θ for spoofing (92.3563°) closely matches with our benchmark of 90°. Furthermore, the corresponding Δ_θ for spoofing agrees with our expectation for practical scenarios which involves sharp turns and changes in travelling directions. For complex paths in figure 7, the values

of Δ_θ for spoofing are even higher (99.8231°, 107.9607° and 141.4167°). On the other hand, the values of len_s for every sample path remains stable within a narrow range ($\approx$ 0.66 to 0.72 m), which is highly comparable to the average adult step length ($\approx$ 0.67 to 0.76 m). Our methodology with the measurement of Δ_θ and len_s shows high competence in spotting location spoofing behaviors, providing convincing identification results in practical situations.

6.3 Practicability

We obtain all the necessary data for our experiment via the Android API (location data from GPS module via location service, device bearings from gyroscope and step count from accelerometer via sensor service). The implementation of the data retrieval process is simple and straightforward by making respective function calls provided in the API libraries. The corresponding procedures are also being well-documented in the Android developer documentation. Due to the frequent updates of the GPS values and sensor readings during the detection process, energy consumption can be a possible concern. A good balance between update frequency and detection accuracy should be taken into careful consideration. Depending on the requirements of the developer, these factors can be easily adjusted.

7 Limitation and future work

Although our detection method shows high competence in identifying location spoofing behavior, the detection accuracy is path dependent. Detection may fail if the travel direction of the simulated path coincidentally matches with facing direction of the device. If an adversary intentionally adjusts the facing direction of his/her device to match with the travel direction of the simulated path when spoofing, he/she may probably escape from the bearing comparison part of our detection algorithm. The introduction of average step length comparison provides an additional dimension in location spoofing detection, but the measurement of step length is user dependent. The step length can variate a lot across people in different age and ethnicity groups, giving uncertainties to the detection results. To improve our detection methodology, additional sensors should be taken into consideration to obtain more information of the movement of the user. The Bluetooth discovery function making use of the Bluetooth chipset equipped in almost every mobile devices nowadays can be a good option to help on retrieving additional information on the surrounding environment of the device so as to determine its current motion status. We will leave it to our future research with further investigation.

8 Conclusion

In this paper, we give clear classification on two common kinds of location spoofing attacks and summarize the corresponding research work in these two areas. We focus on location spoofing attack with fake location reporting and proposed a behavioral detection algorithm on location spoofing in local area by utilizing the gyroscope and accelerometer commonly equipped in most mobile devices. We verify the genuineness of the GPS data by making use of the orientation data and step count provided by the motion sensors. We compare the travel directions deduced from the GPS data and orientation data and check whether the step length of travel matches with the average adult step length to identify abnormal travels. Experiment result shows that there is a notable different on the travel direction between legitimate and spoofing travels ($> 70°$) and the step length calculated in our legitimate travel cases matches with the average adult step length ($\approx$ 0.67 to 0.76 m). Our methodology provides high competence in location spoofing detection and it is simple to implement with a range of strongly supported APIs.

References

[1] Bluestacks official website. `https://www.bluestacks.com/` [Online; accessed on December 15, 2020].

[2] Harry potter: Wizards unite official website.

[3] Jurassic world alive official website. `https://www.jurassicworldalive.com/` [Online; accessed on December 15, 2020].

[4] Magisk manager website. `https://magiskmanager.com/` [Online; accessed on December 15, 2020].

[5] Niantic official website.

[6] Nox app player official website. `https://www.bignox.com/` [Online; accessed on December 15, 2020].

[7] Pokemon go official website. `https://www.pokemongo.com/` [Online; accessed on December 15, 2020].

[8] M. Ahmad, M. A. Farid, S. Ahmed, K. Saeed, M. Asharf, and U. Akhtar. Impact and detection of gps spoofing and countermeasures against spoofing. In *Proc. of the 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET'19), Sukkur, Pakistan*, pages 1–8. IEEE, January 2019.

[9] Y.-H. Chang, Y.-L. Hwang, C.-W. Ou, C.-L. Hu, and F.-H. Hsu. Fake gps defender: A server-side solution to detect fake gps. In *Proc. of the 3rd International Conference on Advances in Computation, Communications and Services (ACCSE'18), Barcelona, Spain*, pages 36–41. IARIA, July 2018.

[10] Y. Fan, Z. Zhang, M. Trinkle, A. D. Dimitrovski, J. B. Song, and H. Li. A cross-layer defense mechanism against gps spoofing attacks on pmus in smart grids. *IEEE Transactions on Smart Grid*, 6(6):2659–2668, November 2015.

[11] Z. Haider and S. Khalid. Survey on effective gps spoofing countermeasures. In *Proc. of the 6th International Conference on Innovative Computing Technology (INTECH'16), Dublin, Ireland*, pages 573–577. IEEE, August 2016.

[12] K. Jansen, M. Schäfer, D. Moser, V. Lenders, C. Pöpper, and J. Schmitt. Crowd-gps-sec: Leveraging crowdsourcing to detect and localize gps spoofing attacks. In *Proc. of the 39th IEEE Symposium on Security and Privacy (SP'18), San Francisco, California, USA*, pages 1018–1031. IEEE, May 2018.

[13] K. Jansen, N. O. Tippenhauer, and C. Pöpper. Multi-receiver gps spoofing detection: error models and realization. In *Proc. of the 32nd Annual Conference on Computer Security Applications (ACSAC'16), Los Angeles, California, USA*, pages 237–250. ACM, December 2016.

[14] G. Liu, R. Zhang, C. Wang, and L. Liu. Synchronization-free gps spoofing detection with crowdsourced air traffic control data. In *Proc. of the 20th IEEE International Conference on Mobile Data Management (MDM'19), Hong Kong, Hong Kong*, pages 260–268. IEEE, June 2019.

[15] F. Restuccia, A. Saracino, S. K. Das, and F. Martinelli. Preserving qoi in participatory sensing by tackling location-spoofing through mobile wifi hotspots. In *Proc. of the 13th IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM'18), St. Louis, Missouri, USA*, pages 81–86. IEEE, March 2015.

[16] F. Restuccia, A. Saracino, S. K. Das, and F. Martinelli. Lvs: A wifi-based system to tackle location spoofing in location-based services. In *Proc. of the 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WOWMOM'16), Coimbra, Portugal*, pages 1–4. IEEE, June 2016.

[17] S. Saroiu and A. Wolman. Enabling new mobile applications with location proofs. In *Proc. of the 10th International Workshop on Mobile Computing Systems and Applications (HotMobile'09), Santa Cruz, California, USA*, pages 1–6. ACM, February 2009.

[18] S. Siamak, M. Dehghani, and M. Mohammadi. Counteracting gps spoofing attack on pmus by dynamic state estimation. In *Proc. of the 9th Smart Grid Conference (SGC'19), Tehran, Iran*, pages 1–5. IEEE, December 2019.

[19] J. R. van der Merwe, X. Zubizarreta, I. Lukčin, A. Rügamer, and W. Felber. Classification of spoofing attack types. In *Proc. of the 26th European Navigation Conference (ENC'18), Gotenburg, Sweden*, pages 91–99. IEEE, May 2018.

[20] J. S. Warner and R. G. Johnston. Gps spoofing countermeasures. *Homeland Security Journal*, 25(2):19–27, January 2003.

[21] S. K. Wong and S. M. Yiu. Detection on gps spoofing in location based mobile games. In *Proc. of the 21st World Conference on Information Security Applications (WISA'20), Jeju Island, Korea*, volume 12583 of

Lecture Notes in Computer Science, pages 215–226. Springer, Cham, August 2020.

[22] B. Zhao and Q. Chen. Location spoofing in a location-based game: A case study of pokémon go. In *Proc. of the 9th International Conference on Advanced Computational Intelligence (ICACI'17), Doha, Qatar*, pages 21–32. IEEE, February 2017.

[23] B. Zhao and D. Sui. True lies in geospatial big data: detecting location spoofing in social media. *Annals of GIS*, 23(1):1–14, January 2017.

[24] B. Zhao and S. Zhang. Rethinking spatial data quality: Pokémon go as a case study of location spoofing. *The Professional Geographer*, 71(1):96–108, September 2018.

Author Biography

Shing Ki Wong received the B.E. and B.B.A. degrees from The University of Hong Kong in 2011 and 2012 respectively. He is a Ph.D. candidate at The University of Hong Kong. His research interests include Big Data Graph Partitioning, Mobile Games Cheat Detection and Location Spoofing Detection.

Siu Ming Yiu is currently a professor in the Department of Computer Science of the University of Hong Kong. He is also an associate executive director of a newly established HKU-SCF FinTech Academy and the associate director of the Center for Information Security & Cryptography. He was named as a Highly cited researcher by Clarivate Analytics in 2016, 2017, and 2019. He is also among the top 1% scholars for 10 consecutive years in HKU (2011-2020). His research interests include cyber security, cryptography, and FinTech.