Identification of device motion status via Bluetooth discovery

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

Abstract

Location spoofing has long been an outstanding issue among mobile devices. By manipulating the devices' GPS values via location spoofing applications, attackers can easily give false reporting on their current locations. Such spoofing behavior can cause serious troubles and even put other people in danger. Many researches are being done in identifying spoofing behavior by making use of the relationship between the location of the device and the signal strength of its connected Wi-Fi network. They give good results in detection on long distance travels while fall short in recognizing spoofing behavior in close range. We address this issue and propose an identification methodology to predict the motion status of a mobile device by taking advantage of the short-range property of the Bluetooth wireless network. We carry out inquiry scans using the Bluetooth discovery functionality available in most mobile devices to collect information of the devices nearby to learn about the surrounding environment, which can then be used to verify the genuineness of the GPS data. Experiment results show that our method can evidently distinguish between various motion status of the device by making statistical analysis on the distribution of appearance of the discovered devices.

Keywords: Location Spoofing, GPS Spoofing, Bluetooth Discovery, Mobile Games

1 Introduction

The application for Global Positioning System (GPS) has become much more diverse in recent years. Apart from offering general location positioning services for traditional industries like the airline companies and the military, it also gives additional assistance in many other different areas. For example, it helps astronomers to improve the accuracy of weather forecasts by projecting temperature levels between the ground and the atmosphere. It also provides a much more efficient way for zoologists to monitor the migration patterns of endangered species. With the rapid development in mobile technology, the use of GPS services nowadays is common and widely spread. Every single mobile device has its GPS module equipped inside to provide location service functionalities to applications for different purposes such as public transports tracking, patients positioning, and even gaming. Although the implementation of location service in mobile devices provides a lot of convenience to the society with a huge amount of location-based applications, it brings out additional difficulties for authorities in validating the genuineness of the GPS data. Mobile users can easily manipulate their GPS locations with a large bunch of existing location spoofing applications in the market. Such kind of false reporting on GPS locations could result in many problems. For example, misreporting on the locations of isolated patients of an infectious disease can hinder the management of the hospital authority and the government, causing potential crisis to the general public. On the other hand, in recent years several popular location-based mobile games such as Pokémon GO [3], Jurassic World Alive [2] and Harry Potter Wizards Unite [1] have been in great success by making use of the GPS data to integrate the game universe with the real

Journal of Internet Services and Information Security (JISIS), volume: 10, number: 4 (November 2020), pp. 59-69
DOI: *10.22667/JISIS.2020.11.30.059*

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

world. The behavior of location spoofing in these location-based games can heavily affect the fairness and gaming experience among players, eventually hurting the game community as a whole.

Nevertheless, only a few measures are being done to prevent cheating with location spoofing in mobile devices. Travel viability check examine the possibility for the user to actually travel between the GPS locations physically, usually making use of constraint between the distance travelled and the travel time taken. Mock location app detection checks whether the usage of location spoofing applications is permitted in the mobile operation system. Malicious app detection identifies and block suspicious applications installed in the device during runtime. Jailbreak and root detection checks if privilege escalation is being done on the mobile device for easy manipulation on GPS data. These prevention measures work well in specific scenarios in countering location spoofing behaviors, but the corresponding hacks against these detection methods are also widely available. Therefore, the problem of false reporting in location data remains unsolvable, gradually becoming a seesaw battle between cheaters and developers.

In this paper, we proposed a statistically based methodology to identify the motion status of a mobile device by making use of the Bluetooth discovery functionality provided by the mobile operating system to verify the genuineness of GPS data provided by the user. We keep track of the frequencies and distribution of appearances of the Bluetooth devices nearby with multiple rounds of discovery to determine whether the host device is in motion. We take advantage of the short-range property of Bluetoooth signals to give proper prediction on the moving behavior of the device in small distances. Our methodology is easy to implement and gives adequate identification results.

The rest of the paper is organized as follow. Section 2 summarizes the research studies made on location-based mobile games and discusses on the related work carried out in location spoofing detection. Section 3 proposes and explains the idea of our detection methodology and discusses the detection criteria adopted in our proposed method. Section 4 discloses and analyses our experiment results. Section 5 points out the limitation of the methodology and suggests possible improvement in future work. Section 6 concludes our work in this paper.

2 Related work

The success and huge popularity on location-based mobile games draws scholar's attention in recent years. Many studies have been done regarding to the novel features and gaming experience of these games. [8] conducts a qualitative survey on players to explore the game experiences on the positive and the negative aspects of Pokémon GO. [9] identifies 10 ways that players violate the rules and regulations of Pokémon GO and gives implications on the design of location-based games. [7] gives lengthy observation and conducts interviews with Pokémon GO players and monitors related online communities to evaluate common types of information sharing that occur in physical and online environments. [6] studies on the behavior changes among Pokémon GO players by conducting online survey and discovers that players are much motivated and positive in their emotions.

At the same time, the abuse of location spoofing in mobile devices for the sake of cheating in location-based mobile games raise the awareness of researchers. [13] encourages cartographers and GIScientists to face the problem of location spoofing head-on so as to develop more effective and trustworthy ways to use geospatial big data by reviewing some common techniques on location spoofing in Pokémon GO. [14] develops a quantitative approach on location spoofing detection by testing it with millions of geo-tagged tweets. It also proposes a qualitative approach to explore the possible motivations and uncertainties on this issue. [15] gives a comprehensive case study based on Pokémon GO to draw scholars' attention to the problem of location spoofing. It examines human and nonhuman agents including gamers, spoofers, game developers, bots, and hackers by applying an actor–network theory (ANT) on them. It argues that location spoofing behaviors should be considered seriously as real human geographic

Table 1: Comparison between existing location spoofing detection methodologies.

Approaches	**Location proof**	**Secure PS system from LSAs**	**Secure LBS from LSAs**	**Fake GPS Defender**
Author	Saroiu *et al.* (2009)	Restuccia *et al.* (2015)	Restuccia *et al.* (2016)	Chang *et al.* (2018)
Verifying agent	Wi-Fi hot spots or cell towers	WiFi hot spots	WiFi hot spots	Edge routers
Peer evaluation	No	Yes	Yes	No
Data retrieval	Client side	Client side	Client side	Server side
Infrastructure modification	Yes	Yes	Yes	No
Implementation scale	Large	Small	Small	Large
Implementation cost	Large	Small	Small	Small

data, rather than simply interpreted as fake data.

However, only a small number of research works has been done on the detection of location spoofing. Most of them make use of the Wi-Fi network to identify spoofing behavior over long distance. [12] make use of a digital certificate called location proof which are signed and issued by access points to the devices upon connection. Such location proof can be verified later by the access points to confirm the current and previous locations of the device. [10] makes use of the idea that devices connecting under the same Wi-Fi address are practically sharing the same physical location so that they can mutually validate each other's current location. It proposed a location validation scheme for participatory sensing (PS) systems without losing any quality of information (QoI) in the system. [11] follows the same idea and proposed a location validation system (LVS) to verify the locations of the users in location-based service systems by filtering out malicious users with multiple rounds of update in their reputations. It suggests that devices are in fact directly connected with each other under the same Wi-Fi range of a mobile hotspot. With a process of mutual verification among the devices, users who are spoofing on their locations can be identified. [5] checks whether the GPS location provided by the device matches the location of the device's edge router and proposed a server-sided and network-based framework to identify location spoofing behaviors. Table 1 gives comparison between the abovementioned location spoofing detection methodologies.

We can see that it is convenient to detect location spoofing with the help of Wi-Fi network due to its huge user base. The location of the routers can be easily identified with the help of existing databases such as WiGLE [4]. Although it works well in detecting location spoofing for long distance, it lacks accuracy in detecting location spoofing in small areas due to the lengthy range of the Wi-Fi signal (up to 46 m indoors and 92 m outdoors for a 2.4 GHz band). Within the range of the same router (an area of up to $\approx 26000\ m^2$), it is difficult to tell whether a device has been moving or not inside that area. Existing methodologies fails to tackle such scenarios effectively and not many studies are being done on location spoofing detection within a small local area. Furthermore, detection on location spoofing in short ranges requires relatively frequent updates on GPS data. Energy consumption could be huge by using Wi-Fi connections in this case. Therefore, alternative approaches with lower energy consumption should be sought to tackle this problem.

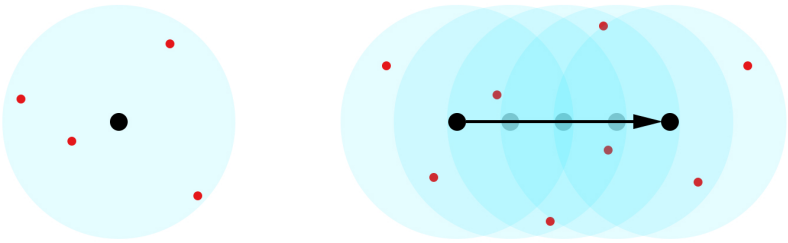

Figure 1: Bluetooth discovery when the host device is stationary (left) and moving (right).

3 Methodology

The Bluetooth technology has been developed for three decades. It provides short range communication between Bluetooth devices using short-wavelength UHF radio waves. It emits signals that travel much shorter than Wi-Fi signals (about 10 times less), while consumes much less power on the other way around (less than 3% of Wi-Fi for the same tasks). In this section, we propose a detection methodology to identify the motion status of a mobile device by making use of the short-range property of Bluetooth signals. We carry out multiple rounds of Bluetooth discovery scanning process to gather information of surrounding Bluetooth enabled devices. We set up three measuring criteria as indicators and perform statistical analysis to determine the motion status of the device.

In order to determine the motion status of a mobile device, we look at the number of times of appearance for each nearby device during the whole discovery process. Figure 1 illustrates two scenarios on Bluetooth discovery. The block dots represent the host device and the red dots are nearby devices. For a stationary device, since the area of each Bluetooth discovery is almost identical, we would expect a significant amount of static devices (e.g. electric appliances, computers, printers) to be discovered in every iteration, with a few other moving devices (mobile phones, headsets, vehicles) appearing from time to time in different iterations. For a moving device, on the other hand, with continuously changes on the discovery area, it is expected that there will be a lot fewer amount of devices, which are usually paired wearables, or other devices moving in the same direction with the host, to be appearing in every discovery. On the contrary, it is expected that there will be a large amount of random devices appearing and being discovered within a few discovery iterations when the host is passing by. Thanks to the short physical range of Bluetooth signals (typically less than 10 m for class 2 devices which most mobile phones belongs to), with a scan time of around 12 seconds in a typical Bluetooth inquiry scan, it is highly unlikely to have a stationary device to be discovered by the host with a common walking speed of around 2 m/s for more than 3 times. With such observation, we can then set up several measuring criteria in predicting the motion status of a mobile device.

3.1 Measuring criteria

3.1.1 Variation of percentage of appearance

To determine whether the host is static or mobile, we keep track on the number of inquiry scans made during the whole discovery process. We divide the number of times of discovery of each discovered device by the total number of inquiry scans to obtain the percentage of appearance for each discovered device. The higher the variation of these percentages is, the more likely the host is in a static environment as there would be a substantial amount of local devices to be discovered in every inquiry interval, with some other moving devices appearing and disappearing when they pass by. The lower this variation is,

the higher the possibility that the host device is in a changing environment, as there is unlikely to have devices to be discovered in every inquiry interval when the host is moving from place to place, while a lot of random devices would be popping up in similar number of rounds when the host device pass by. We define the standard deviation of these percentages as σ_a.

3.1.2 Variation of numbers of discovered devices

The number of Bluetooth devices discovered varies in each round of inquiry scan. It serves as a good indicator on whether or not the host device is moving from place to place. When the host is in motion, the number of devices being discovered in each round would fluctuate a lot more compare to the case that when the host is static. Therefore, it is expected that the variation on the number of discovered devices among the scanning intervals to be much larger when the host is moving, while on the contrary more stable when the host is stationary. We define the standard deviation of the numbers of discovered devices across the discovery intervals as σ_d.

3.1.3 Average of percentages of discovery

We define the *percentage of discovery* of each round of inquiry scan to be the number of devices discovered in that round of inquiry scan in a portion to the total number of devices discovered in all rounds of inquiry scan. That is

$$\frac{\text{number of devices discovered in each scan}}{\text{total number of devices discovered in all scans}}.$$

We calculate the mean of these percentages to indicate how often the host's environment changes (i.e. how fast the host moves). The lower this average is, meaning that more distinct devices are being discovered in the whole period and fewer devices are being discovered in each round, the more likely the host device is in motion and vice versa. We define the mean of these percentages as $\overline{d}$.

3.2 The algorithm

Our algorithm consists of two parts, which first collects devices information during the Bluetooth scanning procedure, followed by statistic calculations on the obtained information after scanning. We identify the Bluetooth devices with their unique Bluetooth MAC addresses and count the number of appearances for each of the devices throughout the scanning process, which will then be divided by the total number of inquiry scans to obtain the percentage of appearance for each distinct device during the discovery process for the calculation of σ_a. We also keep track on the total number of devices being discovered in each scan and the whole discovery for the calculation of σ_d and $\overline{d}$. The details of the algorithm are described as follow.

Table 2: Test cases categorized according to their scenerios.

Scenerio	Description
Walking	outdoor travels in urban areas, indoor travels in malls and complex buildings
Commuting	bus and railway travels
Stationary	sitting in malls, resturants, home and office locations

Algorithm 1: Record device information via Bluetooth discovery and calculate σ_a, σ_d and $\overline{d}$

```
i ← 0 ;
while not end of detection do
    BluetoothDiscovery.start() ;
    while not end of BluetoothDiscovery do
        Devices(i).add(DiscoveredDevice) ;                    // Record devices
    end
    NumberOfDiscoveredDevices[i] ← Devices(i).count() ;
    AllDevices.add(Devices(i)) ;
    i ← i+1 ;                                                 // Next iteration
end
foreach d ∈ AllDevices do
    PercentageOfAppearance[d] ← AllDevices.count(d)/i ;
end
for n ← 0 to i do
    PercentageOfDiscovery[n] ← Devices(n).count()/AllDevices.count() ;
end
σ_a ← var(PercentageOfAppearance) ;
σ_d ← var(NumberOfDiscoveredDevices) ;
d̄ ← mean(PercentageOfDiscovery) ;
```

4 Evaluation

4.1 Experiment setup

We develop an Android application to perform continuous Bluetooth inquiry scans to collect information of nearby discoverable Bluetooth devices. Figure 2 shows the layout of the application. Discovered Bluetooth devices in each scanning iteration are listed on the main area of the screen, with a countdown timer at the top of the interface indicating the remaining time of each experiement. We ignore paired devices which always stick with the host and therefore gives no useful information to the identification process. We carry out multiple scannings with the device moving and being stationary in different occasions (e.g. walking indoor and outdoor, commuting, sitting still at office etc.). Each scanning lasts for 10 minutes, where one third of them are with the device moving (walking), one third of them are with the device being stationary (staying at the same location), and one third of them are with the device being stationary while moving (commuting). Table 2 describes our testing cases and categorizes them according to the three scenerios.

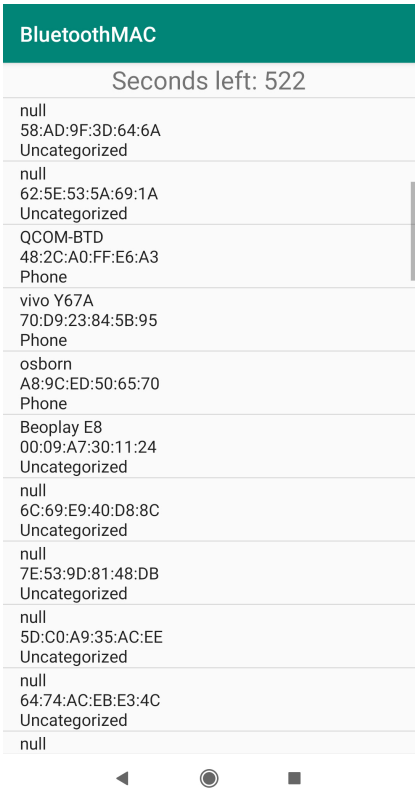

Figure 2: Layout of the Android application for Bluetooth discovery in our experiment.

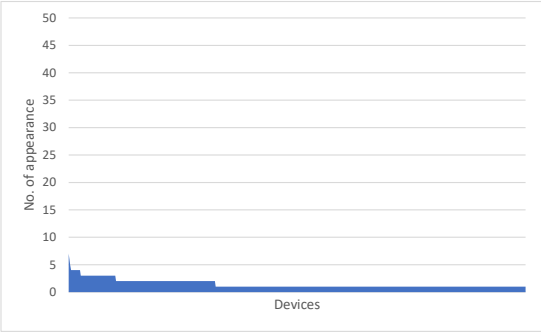

Figure 3: Distribution of appearances of discovered devices when the host is walking.

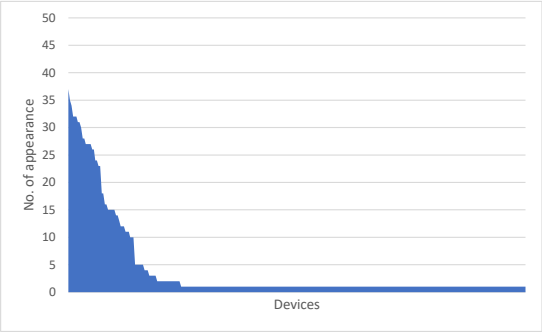

Figure 4: Distribution of appearances of discovered devices when the host is commuting.

4.2 Results

Figure 3, 4 and 5 visualize three sample cases showing the distribution of appearances of discovered devices sorted in descending order when the host is walking, commuting and remaining stationary respectively. We can see that for the case where the host is walking (Figure 3), the numbers of appearance of the discovered devices remain stable at low values throughout the whole scanning process, resulting in a lower value of σ_a. For the case where the host is stationary (Figure 5), there are much more devices being discovered frequently and the numbers of appearance of the discovered devices are evenly distributed

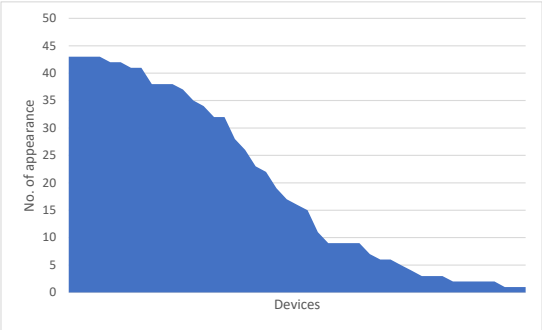

Figure 5: Distribution of appearances of discovered devices when the host is stationary.

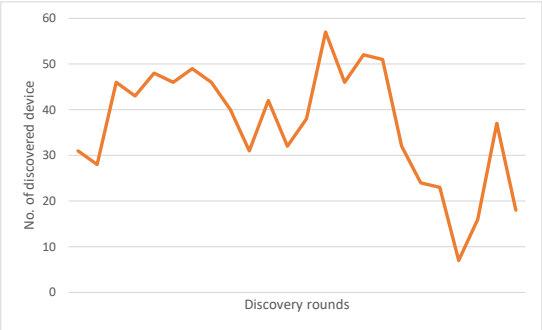

Figure 6: Number of discovered devices throughout whole scanning when the host is walking.

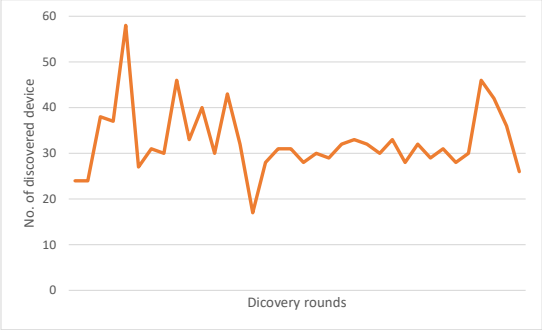

Figure 7: Number of discovered devices throughout whole scanning when the host is commuting.

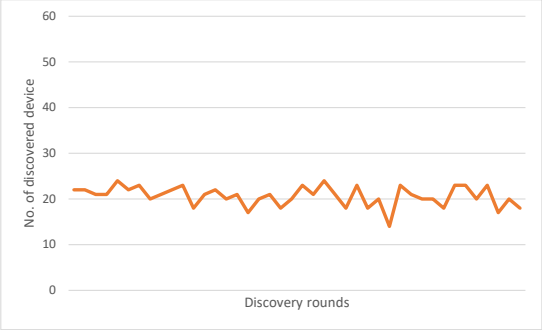

Figure 8: Number of discovered devices throughout whole scanning when the host is stationary.

across the spectrum, resulting in a much higher value of σ_a. The case where the host is commuting lies between the walking case and stationary case, with a small part of devices being discovered frequently and another large set of devices being discovered only for a few times.

Figure 6, 7 and 8 show the number of discovered devices in each iteration throughout the whole scanning process for the three respective sample cases mentioned above. We can see that the numbers of discovered devices fluctuate a lot throughout the scanning process when the host is walking (Figure 6), which will result in a higher value of σ_d. Such fluctuation reduces significantly when the host is stationary (Figure 8), indicating that the host is always discovering the same set of nearby Bluetooth devices, which will give a much lower value of σ_d. Similar to the situation on the appearances of discovered devices, the case which the host is commuting lies between the other two cases (Figure 7), with a relatively stable trend compared to the walking case, while occasionally discovering a huge number of devices in some of the iterations.

Table 3 summarizes our experiment result. We can see that the value of σ_a is much larger when the device is stationary (0.3031) compared to the case when the user is walking (0.0508), such a large value indicates that there are some devices being frequently discovered while at the same time exists a considerable amount of devices being discovered occasionally, making the distribution more disperse. The discovery of devices during walking is relatively more stable and evenly distributed, resulting in a much smaller value of σ_a. The value of σ_a during commutation (0.1952) lies in between, since it is a mixed

Table 3: Values of our measuring criteria at different device status.

Device status	σ_a	σ_d	$\overline{d}$
Walking	0.0508	11.3985	0.0669
Commuting	0.1952	6.3145	0.1086
Stationary	0.3031	3.0499	0.3681

case of the two scenarios abovementioned, where there exists some frequently discoverable devices (e.g. other commuters) while at the same time continuously discovering other new devices while the user is travelling. For σ_d, which indicates the degree of fluctuation of the number of devices discovered, we can see that the value is much lower when the device is stationary (3.0499) compared to the case that when the device is in motion (i.e. walking and commuting) because the environment is much more stable when the device is sitting still. It is lower in commuting (6.3145) compared to walking (11.3985) due to the static environment inside the transport. The value of $\overline{d}$ is much higher when the device is stationary (0.3681) due to the fact that only very few new devices are being discovered in a stable environment, compared to the other two cases (0.0699 and 0.1086) which the device is continuously discovering new devices while moving.

4.3 Practicability

Our methodology requires the Bluetooth function to be enabled in the host device for scanning purposes. It cannot be a concern as most of the mobile users nowadays would have their devices' Bluetooth module turned on for various reasons such as connecting with headsets or making payments with the NFC technology. On the other hand, the existence of other surrounding mobile devices could degrade our measurement results. However, since the device discoverability is being disabled by default in most of the mobile operation systems, it should pose minor effect to the scanning results. Last but not least, long period of continuous scanning using Bluetooth signals could cost vast energy consumption to the device but fortunately, with the adoption of the Bluetooth Low Energy (BLE) technology, such problem can be adequately alleviated.

5 Limitation and future work

Although our methodology is capable to give clear indication on the motion status of a device, it may not work in some marginal cases (e.g. when the host is sitting still in an open area with no static Bluetooth devices nearby but a lot of mobile devices passing by, or when the host is moving with a lot of other mobile devices together in the same direction coincidentally). More information from different perspectives is needed in order to give better identification in such scenarios. In fact, our research only serves as a shallow opening on this issue. As there are much more additional information can be retrieved during Bluetooth discovery, more precise and accurate motion status identification can be possibly done. For example, one can deduce from the name, Bluetooth type and class of the device to predict the actual category of the device and therefore makes better estimation on the location of the host device. What's more, one can also make use of the Received Signal Strength Indicator (RSSI) to predict the distance between the host and nearby devices to get extra information on the surrounding environment. The analysis of these data is untrivial and we will leave it to our future work with further investigation.

6 Conclusion

In this paper, we proposed an identification methodology by making use of the Bluetooth discovery functionality in mobile devices to predict the motion status of the device. We identify each distinct Bluetooth device with its Bluetooth MAC address and record the number of appearances for each of the devices. We also keep track on the number of devices discovered in each round of discovery. We make a statistic approach to deduce the behavior of the device from the data collected. Experiment results shows that our methodology can efficiently differentiate between different motion status of a mobile device. Energy concern could be a problem while it can be efficiently alleviated with the help of the advancing BLE technology.

References

[1] Harry potter: Wizards unite official website. `https://www.harrypotterwizardsunite.com/`, [Online: Accessed on October 20, 2020].

[2] Jurassic world alive official website. `https://www.jurassicworldalive.com/`, [Online: Accessed on October 20, 2020].

[3] Pokemon go official website. `https://www.pokemongo.com/`, [Online: Accessed on October 20, 2020].

[4] Wigle: Wireless network mapping. `https://wigle.net/`, [Online: Accessed on October 20, 2020].

[5] 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.

[6] T. Kari, J. Arjoranta, and M. Salo. Behavior change types with pokémon go. In *Proc. of the 12th International Conference on the Foundations of Digital Games (FDG'17), New York, New York, USA*, pages 1–10. ACM, August 2017.

[7] J. H. Lee, T. Windleharth, J. Yip, and M. Schmalz. Impact of location-based augmented reality games on people's information behavior: A case study of pokémon go. In *Proc. of the iConference 2017, Wuhan, China*, pages 459–468. iSchools, March 2017.

[8] J. Paavilainen, H. Korhonen, K. Alha, J. Stenros, E. Koskinen, and F. Mäyrä. The pokémon go experience: A location-based augmented reality mobile game goes mainstream. In *Proc. of the 2017 CHI Conference on Human Factors in Computing Systems (CHI'17), Denver, Colorado, USA*, pages 2493–2498. ACM, May 2017.

[9] J. Paay, J. Kjeldskov, D. Internicola, and M. Thomasen. Motivations and practices for cheating in pokémon go. In *Proc. of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI'18), Barcelona, Spain*, pages 1–13. ACM, September 2018.

[10] 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 2015 IEEE International Conference on Pervasive Computing and Communication Workshops, St. Louis, Missouri, USA*, pages 81–86. IEEE, March 2015.

[11] 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.

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

[13] 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.

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

[15] 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 includes Big Data Graph Partitoning, 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.