

Review Article

Sensors of Smart Devices in the Internet of Everything (IoE) Era: Big Opportunities and Massive Doubts

Mohammad Masoud ¹, Yousef Jaradat,¹ Ahmad Manasrah,² and Ismael Jannoud¹

¹Electrical Engineering Department, Al-Zaytoonah University of Jordan, Amman 11733, Jordan

²Mechanical Engineering Department, Al-Zaytoonah University of Jordan, Amman 11733, Jordan

Correspondence should be addressed to Mohammad Masoud; m.zakaria@zuj.edu.jo

Received 12 November 2018; Revised 12 January 2019; Accepted 11 February 2019; Published 15 May 2019

Academic Editor: Matthew Brodie

Copyright © 2019 Mohammad Masoud et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smart device industry allows developers and designers to embed different sensors, processors, and memories in small-size electronic devices. Sensors are added to enhance the usability of these devices and improve the quality of experience through data collection and analysis. However, with the era of big data and machine learning, sensors' data may be processed by different techniques to infer various hidden information. The extracted information may be beneficial to device users, developers, and designers to enhance the management, operation, and development of these devices. However, the extracted information may be used to compromise the security and the privacy of humans in the era of Internet of Everything (IoE). In this work, we attempt to review the process of inferring meaningful data from smart devices' sensors, especially, smartphones. In addition, different useful machine learning applications based on smartphones' sensors data are shown. Moreover, different side channel attacks utilizing the same sensors and the same machine learning algorithms are overviewed.

1. Introduction

Internet of Everything (IoE) is an information technological term that combines sensing, computation, information extraction, and communication functionalities together in a device. IoE allows different electronic devices with different capabilities to sense the environment and to communicate for data exchange [1]. IoE is the general form of wireless sensor networks [2]. IoE nodes may have different classes, types, and capabilities. For example, smartphones, tablets, laptops, home appliances, and even cars are examples of nodes in IoE. These nodes can sense the environment utilizing their different sensors and process data, retrieve useful information, communicate over the Internet, and control their behavior adaptively. IoE nodes' smartness and intelligence are not in their computational capacity, but in their ability to communicate and exchange information. Communication links allow these devices to learn from their sensed data. It trains these devices to leverage its information to perform new useful tasks [3]. For example, a fridge with an embedded processor is not smart until it has the ability to communicate

with people, other fridges, and supermarkets to order missing items. Moreover, it should select from different supermarkets to buy the items with price offers. This smartness came out from data communication over the Internet.

IoE is a complex approach with massive applications, dreams, and myths. It has uncountable applications in health, engineering, computer science, marketing, and even social sciences [4, 5]. However, it has many issues that require more investigation. Security and privacy dominated in the IoE research field [6]. How to secure your data and applications is a hot research topic in IoE. However, people security as a drawback in the IoE paradigm should be studied. Many questions emerged in this field. What to sense from the environment and what to upload to the Internet? How to enhance privacy if sensors are everywhere in people's lives? How to teach people to deal with IoE in a responsible way? Can IoE be harmful?

Smart devices play a main role in IoE [7]. They are equipped with multicomunication interfaces, such as Wi-Fi, Bluetooth, near-field communication (NFC), and cellular communication. In addition, they are equipped with

a massive number of sensors. Moreover, they have embedded operating systems (OSs) that are referred to as IoT OSs [3]. When smartphones are mentioned in this survey, we are referring to smartphones, tablets, and smartwatches since they have the same characteristics with few industrial differences. According to the statistics reported by Statista (<https://www.statista.com/statistics/330695/number-of-smartphone-usersworldwide>), the number of smartphones worldwide exceeded 2.8 billion with an estimation of 5 billion in 2019. Smartphones have been employed heavily in controlling and monitoring the process of hundreds of smart home products. For example, WeMo (Belkin Wemo: home automation, <http://www.belkin.com/us/Products/home-automation/c/wemo-home-automation/>) product allows the users to control multiple features in their houses, such as power usage of different appliances. This product is controlled by smartphones. Another example is Apple HomeKit (<https://developer.apple.com/homekit/>) for security and surveillance systems. A third example is Reemo (<http://www.getreemo.com/>) that converts houses into smart homes. Smartphones play a monitoring and controlling role in these applications. However, smartphone capabilities and sensors allow them to play a greater role in health, identification, localization, and tracking.

Sensors are used to enhance the smartphones' usability. However, researchers and developers attempted to leverage these sensors in much more complex applications, such as user identification, subscriber tracking, and even personality traits. These applications require mining of hidden information of the smartphones' sensor data. In other words, sensor data are leveraged in new indirect ways to predict and estimate new features not directly designed to be assessed by these sensors. This new usage paradigm of smartphone sensors reveals privacy and security issues since smartphone users are willing to upload their harvested data without any awareness of the information that can be mined from them [8]. This issue was referred to as *big data accident* [9]. The author in [10] proposed a system based on normal accident theory to show drawbacks of big data accident. He has shown that big data may be converted to "evil" in mining free uploaded information. In [8], the author has shown that users have limited control over the uploaded data which is one of the main privacy concerns in IoE. In [9], the authors proposed ten rules to guide the privacy and security issues in big data and the ethics that should be emerged. The main motivation in this work is to gain more insights into the privacy issues of smartphones as devices in IoE.

In this work, some of the interesting applications that have been proposed and designed for exploiting smart devices' sensor data are shown as the big opportunities in the new era of IoE. Nevertheless, the accuracy of these applications is shown as one of the substantial issues that requires answers. On the other hand, security and privacy issues are introduced as the doubts of these devices. In this work, we seek to show that security, privacy, and big data accuracy of smart devices in the era of IoE are data content stored not only in the device but also in Internet servers. However, even the raw data extracted from smart device sensors can introduce more threats than the stored contents.

Our contribution in this work is summarized as follows:

- (i) Surveying the applications of smart devices' sensor hidden data that have been conducted over the period of 2004–2018
- (ii) Dividing the threats of smart devices' sensor hidden data into three main categories and proposing different scenarios of these threats
- (iii) Discussing several proposed solutions for the hidden data threats
- (iv) Proposing a simple approach to start inferring hidden information from smart devices' sensor data without deep programming skills

The rest of this paper is organized as follows: Section 2 overviews smart device architecture and their internal components. Section 3 shows how data mining and IoE collide in the area of smart devices. Section 4 shows the useful applications of hidden data extraction. Section 5 shows the disadvantages of extracting sensor hidden data and the methods to start digging the smart device hidden data. Finally, we conclude this work in Section 6.

2. Smart Device Architecture

Smart devices in this work are defined as the hand-held devices. These include smartphones, tablets, and smartwatches. These devices have approximately the same internal architecture with differences in the speed, size, number of sensors, and storage capacity. In addition, they adopt the same operating systems and software stacks. The apps designed for a smartphone work and operate in tablets. Figure 1 shows the block diagram of the internal architecture of a smart device. As shown in the figure, smart devices have two main parts: processors and sensors. There are also other interfacing parts that connect the sensors to the processors, such as analog-to-digital convertors (ADC), digital-to-analog converters (DAC), voice codecs, and the main memories to handle smart devices' app instructions. The following sections overview the main part of the figure with emphasis on sensors.

2.1. Smartphone Processors. Modern smartphone architecture contains two or more processing units. These include application and baseband processors. In the following, these processors are introduced.

2.1.1. Application Processor. This processor is similar to a central processing unit (CPU) in personal computers (PCs) or laptops. Nevertheless, it has three main design features. First, it has a power saving mechanism. Second, it is responsible for managing all sensors, SD card, and communication modules of the smartphone. The embedded sensors in the smartphone are analog sensors. These sensors require analog-to-digital converters (ADC). To fit all of these components in slim smartphones, system on chip (SoC) technology is utilized as in microcontrollers. Moreover, microelectromechanical system (MEMS) technology is

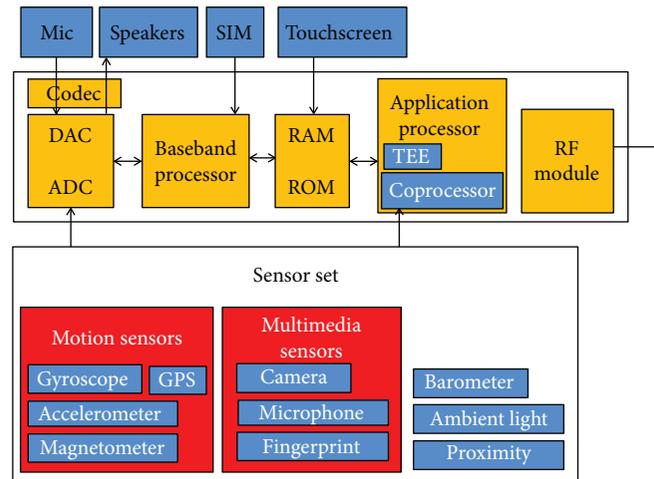


FIGURE 1: Smart device internal architecture.

utilized to design the small sensors. Third, it utilizes the trusted execution environment, which is responsible for storing the data in a trusted, secure, and protected area [11]. In addition to these components, another coprocessor may be embedded, such as Huawei Kirin 970, Apple M7, and Motorola X8. This coprocessor is a low-power electronic component that has its own framework and is capable of natural language processing (NLP) and contextual computing processing (CCP). CCP processes accelerometer, gyroscope, and magnetometer sensor data in real time. The coprocessor is always on and processes sensor data in real time even if the application processor is in a low-power mode and the touchscreen is turned off. The Google audio search “Google Go” in Android smartphones is a good example of NLP capabilities. New coprocessors have neural network capabilities, such as Huawei Kirin 970, which can be found in Huawei Mate 10.

2.1.2. Baseband Processor. This processor is a hardware isolated component that has connections with subscriber identity module (SIM) cards, microphone, and speakers. It is responsible for the cellular communication, SMS, and data over the cellular network. It is equipped with real-time operating systems (RTOS). This processor is isolated to allow voice calls to continue in a normal way even if the other components and applications of the smartphone are overloaded. Finally, this processor is responsible for the handoff process between cellular network cells. It is worth mentioning that all of these processors may be designed in the SoC method to allow shared memory access.

2.2. Sensors. Smart device sensors have been embedded in these devices to enhance their usability, controllability, and management. For example, the proximity sensor has been added to enhance the power management of the device; i.e., if the device is near the user’s ear, the screen will automatically turn off. Another example is an accelerometer that senses screen positioning and rotates its content according to users’ positions. And the final example is the battery sensor

that controls the charging process and the temperature of the battery.

Hidden data research has shown that the data sensed from these sensors can be utilized and interpreted to show other information as in the following sections. Moreover, Section 3 shows how the communication and networking parts equipped in smart devices can be leveraged as hidden data-harvesting sensors. This leads to the categorization of smart device sensors according to their functionalities into active and passive sensors. Any sensor may act as an active or a passive sensor according to its usage. In other words, if the data harvested from a sensor is leveraged in the same way as the smart device designers or developers designed it, it is called an active functionality. However, if the collected data has been interpreted in new ways, these sensors are functioning in a passive way. If the sensors are leveraged in this way, hidden information problem occurs. In the following sections, different smart device sensors are introduced.

2.2.1. Touchscreen. Touchscreen is an electronic component that is responsible for the basic input and output operations. It is used for tapping and character typing. Three main interaction procedures are defined for touchscreen. First, touching or tapping is defined as the process of clicking on the screen in any location to open, to close, or to type a character. It is the main activity of the touchscreen. Second, multitouch is defined as the process of tapping the screen by more than one finger simultaneously. This function is heavily used in gaming applications [12]. Third, gesture is defined as the process of drawing a certain pattern on the touchscreen. Gestures may be implemented with one finger as drag and drop or multifingers as in the process of resizing photos and changing camera zoom. Many research and development works have been conducted to exploit the data of these three activities in different methods to obtain some hidden data. One of the visualization methods of touchscreen data is heat maps.

(1) *Heat Maps.* One of the new data visualization methods of multitouching or gesture on a smartphone screen is known as

heat maps [13]. Developers have developed multiple methods to generate these maps [14]. Figure 2 shows an example of these maps.

These maps as mentioned earlier are used for data visualization purposes. Many smartphone applications have been written to utilize these maps to debug written applications and study user behaviors when debugging application issues, such as Appsee [15]. Moreover, many works have been conducted to study touch gesture utilizing touch maps for health diagnoses, such as Down syndrome [16], perceived difficulty [17], and issues in fine motor skills and eyes [18].

(2) *Touchscreen as A Passive Sensor*. All the examples that we will show utilize the touchscreen in an active way: touching speed, delay, typing time, and gestures. However, researchers found another method to obtain useful data from the touchscreen that can be utilized with other smartphone sensors to study sleeping behaviors of the users by counting how many times the touchscreen opens and closes [19]. Moreover, it can be utilized with the alarm application to study how fast users respond to wake-up alarms [20].

2.2.2. *Motion Sensors*. Three main sensors are embedded in modern smart devices for motion detection: accelerometer, gyroscope, and magnetometer. The accelerometer detects changes in the device displacement, orientation, and tilt around three axes by measuring acceleration forces. Its operational theory depends on the value changes of capacitance while a movable mass freely moves between the fixed plates in the MEMS. The total voltage changes from all plates can be recorded and utilized. Figure 3 shows the simple 2D structure of the accelerometer.

On the other hand, the gyroscope measures how fast the device rotates along the three axes [21]. Its internal structure is similar to the structure of the accelerometer. However, the rotational power moves the mass to change the capacitance values of the internal fixed plates. Figure 4 shows the simple 2D structure of the gyroscope. In fact, gyroscopes and accelerometers are often used together in applications as shown in Section 4.1.

A magnetometer is a sensor that measures the strength of the magnetic field around the phone from which the phone is able to obtain its absolute direction related to the earth's geo-magnetic field [22]. Most magnetometers depend on the amount of voltage that is detected across a metallic element when a magnetic field is present. Therefore, magnetometers are mainly used in electronic compass applications [23, 24].

Motion sensors are analog sensors. The output of these sensors is a varied level of voltage. The voltage variation is converted using ADC into a digital number that can be read and shown in the digital world. Motion sensors have different frequencies, which define how many new measurements are taken every second. To extract useful information of motion sensor data, features are extracted. To extract these features, the frequency of reading is set. Moreover, multi-reading values are grouped together to form a window. The size of these windows varied in the conducted research from 10 to 120 readings. Finally, different features are



FIGURE 2: Heat maps (<https://uxcam.com/features/touch-heatmap>).

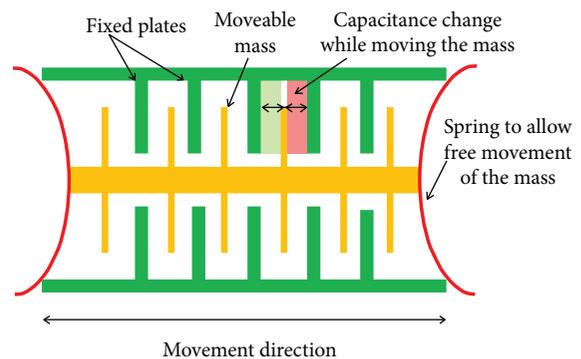


FIGURE 3: Accelerometer internal structure.

calculated from these windows. These features are categorized in three main classes: time, frequency, and wavelets. Table 1 shows the most popular time-domain features, and Table 2 shows the frequency-domain features that are dominant in hidden information extraction from these sensors. The definitions of these features and their equations can be found in [25].

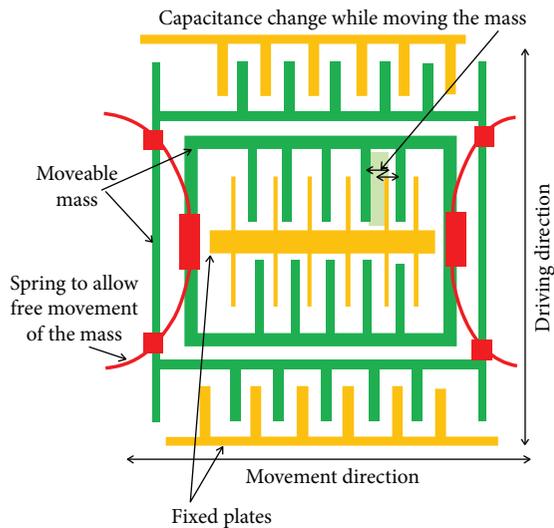


FIGURE 4: Gyroscope internal structure.

2.2.3. Multimedia Sensors. Two main multimedia sensors are embedded in smart devices: camera and fingerprint and microphone. In the following sections, the camera image acquisition process and fingerprint sensors are introduced.

(1) *Camera.* Shooting a photo with a smart device camera passes five different complex stages. The process starts by collecting the light through the camera lens and focusing the light on the internal filter. Subsequently, the output RGB colors are passed to the main camera sensor, the CCD/CMOS sensor. In this stage, each color is manipulated as separated components. To view the last image, color interpolation and image postprocessing step are required. Each one of these stages leaves a fingerprint on the obtained image. This glitch may be utilized to track any photo back to the camera that took it as shown in the following sections. Figure 5 shows the image harvesting pipelining procedure of a smartphone camera.

(2) *Fingerprint.* Fingerprint is a type of biometric recognition systems. Biometric recognition can be defined as the process where the identity of the user is established through identification or verification [26]. It gains popularity since its process depends on the users as who they are and not something they carry or remember like other traditional security systems. The biometric recognition feature heavily depends on the physical, chemical, and behavioral characteristics of the users' body like the fingerprint, iris, face, voice, or even body odor or body heat [27]. Delac and Grgic conducted a nice survey of biometric recognition methods covering most of them [28]. Among those defining characteristics, the fingerprint is the most commonly used in user identification systems since users have distinctive fingerprint patterns for each finger [29]. Hence, fingerprint systems are basically pattern recognition systems for the fingers [30] where the sensor measures the distances and detects the patterns between the bumps and grooves that shape the fingerprint [31]. After

that, the system either compares the result with the biometric data that were previously acquired from the user—verification process—or compares it with a database of fingerprint biometrics from different users—identification process [26].

There are two main types of fingerprint sensors that are still popular and widely used in different biometric recognition systems [32, 33]: optical sensors, where the light that reflected off the fingerprints' ridges and valleys is captured and a fingerprint image is created [34, 35] as shown in Figure 6, and capacitive sensors, where the same procedure is done utilizing the capacitance differences in the fingerprint to create the same image as illustrated in Figure 7. The focus here will mainly be on the capacitive sensors since almost all smart mobile phones that offer the biometric recognition are equipped with capacitive fingerprint sensors. This user identification method is becoming more and more popular among mobile phone users. In fact, studies have shown that around 35% of people use the fingerprint recognition as a user verification method on their phones [36]. It has been estimated that more than half of the mobile phones that will be sold in 2019 will be equipped with fingerprint sensors [29, 37]. And even though the fingerprint can be considered a secure way of locking and unlocking a mobile phone, there are some techniques and methods that may be used to create fingerprint spoofs to hack or unlock a mobile phone. Cao and Jain showed that a smartphone can be hacked or unlocked successfully using a 2D printed fingerprint from the original user [29]. Other studies went even further by constructing a fingerprint image from minutiae. The results showed that there is a very high resemblance between the original and the reconstructed fingerprint [38]. Ben-Asher et al. suggested a two-step authentication method where the fingerprint, combined with the touchscreen, is used to verify or identify the user [39].

2.2.4. Barometer. A barometer is one of the sensors that are recently added to smartphones. It measures changes in the atmospheric pressure in the surroundings of the phone. It is very sensitive since it can measure changes in atmospheric pressure inside the same building or structure. It can be utilized to predict weather. Moreover, it can measure the altitude of the device [40]. Wu et al. have shown that smartphone barometers can be used to detect the buildings' door opening/closing events anywhere inside the building based on sudden changes in atmospheric pressure readings [41].

2.2.5. Ambient Light Sensor. An ambient light sensor is a photodetector sensor that detects the surrounding or ambient light of the smart device and reconfigures the brightness of the smart device screen. It is also utilized to dim the screen to reduce power consumption of the battery. In [42], it has been utilized to study the mental health of smart-watch users. Moreover, it will be shown in Section 4 that this sensor has been widely leveraged to extract users' screen locking patterns.

TABLE 1: Time-domain features.

Feature	Citation	Definition
Mean	[18, 74, 75, 92, 95, 180–184]	The summation of data points divided by their number
Std deviation	[18, 58, 74, 75, 180, 183, 184]	It is the square root of variance
Average deviation	[18, 58, 74, 180, 183]	The average separation of data points from their mean or average value
Skewness	[18, 180, 183]	Measures the asymmetry from the mean value. It utilizes the mean and the variance
Kurtosis	[18, 180, 183]	Estimates the frequency of extreme values. It utilizes the mean value in its formula
RMS amplitude	[18, 180, 183, 184]	It is leveraged to calculate the power of a signal. It utilizes the maximum value of a set
Lowest value	[18, 180, 182–184]	The maximum data point
Highest value	[18, 180, 182–184]	The minimum data point
ZCR	[18, 183]	Zero crossing rate is a counter of how many times the data points cross the zero value
Nonnegative count	[18, 183]	Total number of positive data points in a set
Average absolute difference	[75, 185]	The average of the total differences between all data points in a set
Time between peaks	[75, 184, 185]	The number of points between two high peaks or low peaks
Binned distribution	[75]	The processes of grouping data points into smaller number of points or “bins”

TABLE 2: Frequency-domain features.

Feature	Citation
Spectral centroid	[18, 180, 181, 183]
Spectral Std deviation	[180, 183]
Spectral kurtosis	[18, 180, 183]
Spectral skewness	[18, 180, 183]
Spectral crest	[18, 180, 183]
Irregularity-J	[18, 180, 183]
Smoothness	[18, 180, 183]
Flatness	[18, 180, 183]
Roll off	[18, 180, 183]
Entropy	[18, 183]
Brightness	[18, 183]
Roughness	[18, 183]

2.2.6. *Other Sensors.* There are other sensors embedded in smart devices, such as proximity and battery temperature sensor. However, few applications associated with these sensors are found in the literature. The battery temperature sensor has been used in health applications to tackle death situations when the body temperature drops rapidly [43]. For the proximity sensors, to the best of our knowledge, no applications or research has been conducted to infer different information from its harvested data.

3. Smartphones, Data Mining, and IoE

Data mining is the science of digging useful information from big data records and repositories. These repositories are created from user contents and machine sensors. The issue is not how to harvest these data. The issue is how to mine it. Smartphones are equipped with tens of sensors and

electronic components that generate data in real time [44]. These electronic components and sensors have been embedded in smartphones to enhance usability of these devices. However, researchers have found massive methods to leverage these components and sensors to obtain different information. Many open-access datasets have been collected over the years. They can be downloaded freely from the Internet. One of these datasets is the LiveLab dataset [45], which consists of mobile logs of 100 volunteers over the 14-month period. The dataset consists of fifteen different SQL tables. It has been studied in more than 278 scientific papers according to Google Scholar. Different hidden information has been extracted from it. Another online available dataset is in [46] which has been cited in 342 papers. 30 volunteers participated to collect it. It focused on the accelerometer sensor. It has been extended in [47] and obtained another 130 citations. They extended it by more instances. However, they did not add more sensors. Another example of a dataset that has been collected is in [48]. This dataset focused on the Wi-Fi module in the smartphone, accelerometer, and gyroscope. Moreover, smartwatch data has been also recorded. A final example is the massive dataset [49, 50], which consists of life-logged data of 35 users over two months. It recorded all smartphone activities of the users. This dataset obtained approximately 100 citations. A common feature of all of these datasets is that they did not record any of the users’ content or any private data. In other words, the data collected are treated as normal data from smartphone users. With more than a thousand paper published with different extracted information from these datasets of nonprivate contents and data, it is obvious how this nonprivate data led to the extraction of massive information that can track and identify users’ activities.

As mentioned, the problem is not in harvesting the data itself. The real problem is how to connect the data from different sensors to focus on another hidden meaning. The

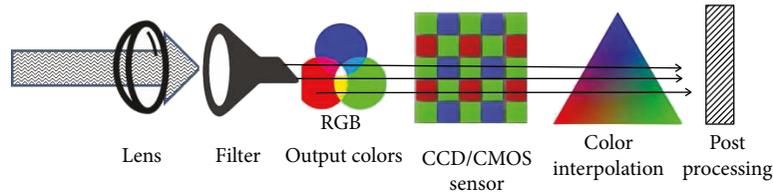


FIGURE 5: Camera image harvesting pipeline.

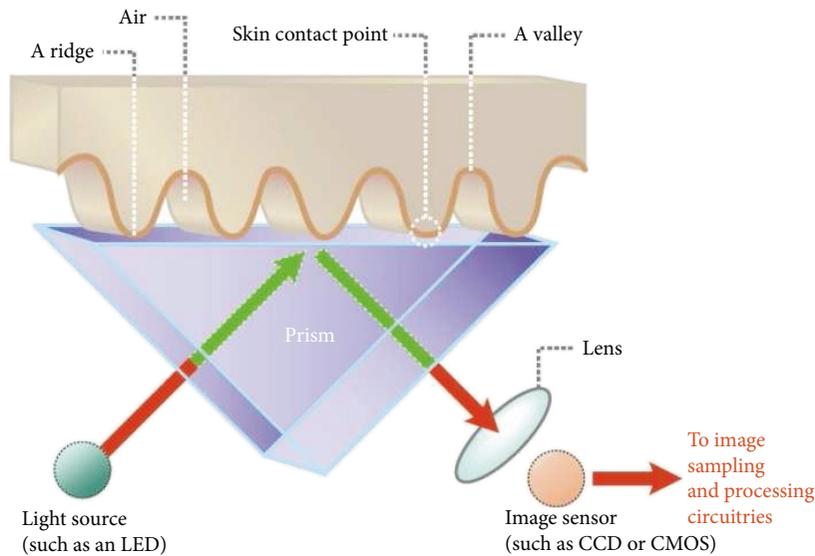


FIGURE 6: Optical sensor fingerprint (<https://www.androidauthority.com/howfingerprint-scanners-work-670934>).

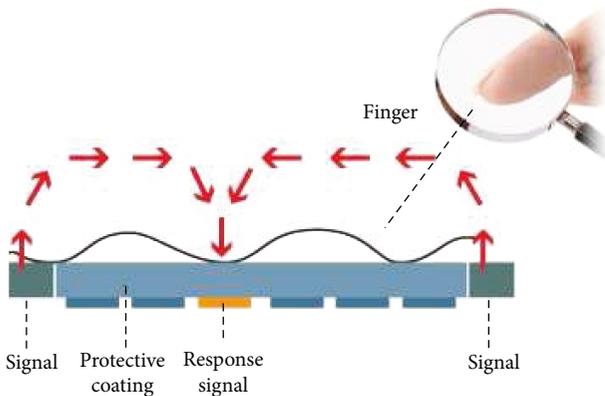


FIGURE 7: Capacitive sensor fingerprint (<http://biometrics.mainguet.org/types/fingerprint/fingerprint-sensors-physics.htm>).

mining process is not also an issue; machine learning algorithms are useful in finding models for the required focused information [44]. This process is like hacking a system. Information is harvested from active and passive probing, such as the sensor data. Subsequently, mining is leveraged to find errors, breaches, and bugs in the system. Finally, algorithms are written to exploit the system. The hard step in the data mining for the big data system is to connect the inputs. In other words, extract useful features from the data and to find information of the harvested data.

Machine learning algorithms (MLAs), supervised and unsupervised, are used heavily in different well-known applications, such as spam filtering, expert systems, and friend suggestions in a social network. Many programming libraries in all programming languages have been written to allow the implementation of MLA in few lines. This allows researchers to focus on the developed application and the interpretation of the data. Figure 8 shows the most popular MLA utilized in the conducted smart device sensor hidden data extraction works. As shown in the figure, the number of these algorithms is massive and they cannot be introduced in one paper. However, three main algorithms will be introduced in the next sections: random forest, support vector machine (SVM), and artificial neural network (ANN). These algorithms have been selected since they have been leveraged in more than 70% of the conducted research surveyed in this paper.

3.1. Random Forest. Random forest is a supervised MLA that has two main applications, regression and classification. Random forest is an enhanced version of decision trees that have been introduced in the 80s. In random forest, multidecision trees are constructed from the same training data. Subsequently, these trees are averaged to obtain the required output. Random forest has been proposed to tackle two main issues in the classical decision trees, overfilling and high variance [51]. As in decision trees, random forest utilizes the “bagging” training method to reach stable and accurate output.

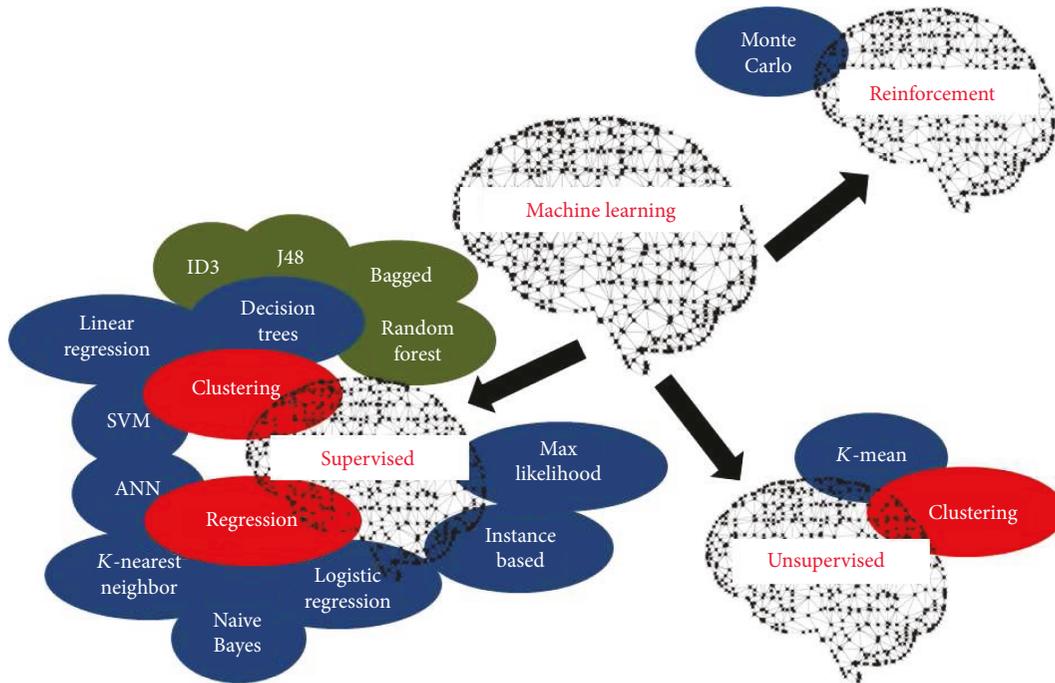


FIGURE 8: MLA in the smart device data extraction process.

3.2. Support Vector Machine (SVM). Like random forest, SVM is a supervised MLA that can be used for regression and classification. However, SVM classification has dominated. In SVM, data points are plotted virtually in the feature dimensional domain and a seeking process of a hyperplane that separates these points into multiclass is initiated. This process is initiated by selecting a number of support vectors from the harvested data. SVM is an enhanced version of logistic regression MLA, where multiclass can be obtained.

3.3. Artificial Neural Network (ANN). Like random forest and SVM, ANN is a supervised MLA that can be used for regression and classification. ANN has many types and classes. The easiest and the commonest type is multilayer feedforward networks. In this type, different numbers of nodes are utilized in three main layers, input, output, and hidden layers. This type is an enhancement of logistic regression. SVM and ANN are similar in many technical areas. However, ANN has a fixed number of hidden nodes in the hidden layers and a fixed number of nodes in the first layer equal to the number of features plus a bias. On the other hand, SVM selects a number of data from the training data to be the support vector. This means that the number of nodes in SVM is not fixed. Moreover, ANN support multioutput unlike SVM that supports single output.

4. Hidden Information Inferring Applications

In the following, an overview of five main applications of smartphones' sensor inferred data is investigated. The accuracy of the extracted information is inspected in Section 4.2.

4.1. Applications of Smartphone Sensors. Some of following applications have been surveyed in [22]. Each one of these applications will be shown with some examples of the conducted works in the field.

4.1.1. Keystroke Authentication. Keystroke authentication (KA) is a set of methods and tools that authenticate a user of a computer or a smartphone through the user's behavior. Thousands of research papers have been written to show how different features of touching patterns can distinguish users. One of the first attempts to study KA using keyboards is [52]. The authors attempted to study KA statistically. 15 different users have been requested to type a sentence of 43 characters 11 times. Five different features have been harvested and compared: key pressing duration, relative keystroke speeds, relative key pressing order, Shift key, and its classes. Key pressing duration was the first to be studied. They observed that the behavior of the same users in typing the same sentence for 11 times did not change; however, it varies across different users. However, the most effective feature in defining the users is the key pressing speed.

These attempts have been carried out in a smartphone. In [53], an Android-based smartphone has been used to collect the touching pattern of 20 users. Three main data columns have been collected: actions (pressing down and pressing up) and screen location. 21 different features have been extracted from these collected columns. Two machine learning classifiers have been evaluated: ANN and the proposed optimized PSO-RBFN. It has been observed from the results that the normal ANN achieved more than 93% accuracy. In [54], seven ML algorithms have been compared for KA. A string consisting of 664 characters has been used. Three different data columns have been recorded: character, key hold

duration, and system time. Subsequently, features have been constructed from these data using the N -grams technique. 4-, 3-, and 2-grams features have been constructed. The results show that higher N -grams obtain better results and fewer errors. Finally, in [55], the authors proposed a KA for smartphones based on four different features: hold time, intertime, distance (between two different pressed characters (in pixels)), and speed. Moreover, they selected these features after studying and categorizing KA features into three main classes. The first category is the way users type a message in the touchscreen where data has been harvested to extract features. Different works have been conducted using this style [55–57]. Different ML algorithms have been compared. The extracted results are promising. The second category utilizes motion sensors with the touchscreen as in [58]. The final category is gesture-based as in [59, 60]. It is worth mentioning that for the popularity of KA in smartphones, many surveys have been written [61–63]. Moreover, in [61], the popularity of KA research and paper publishing has been shown.

In addition to smartphones, smartwatch KA got intensive popularity in the past years. In [64], the authors utilized the motion sensors of a smartwatch with time-domain features to authenticate users. The KNN algorithm has been applied on 20 users. An accuracy exceeding 80% has been reported. In [65], a continuous real-time user authentication system has been proposed utilizing the smartwatch and neural network algorithm. In [66], a system has been proposed to use KA to unlock a smartwatch based on hand waving patterns. Other examples have been surveyed in [64]. Another accelerometer work has been conducted in [67] for device-to-device authentication when connecting headsets and smartwatches to smartphones.

KA methods can be summarized in three steps. First, multiple features are harvested from user inputs, such as typing speed, delay time between different characters during tapping, and multitouch usage [68, 69]. Subsequently, these features are normalized and converted into a matrix of input features and output results. Finally, these data are fed into a machine learning algorithm, such as artificial neural network (ANN), support vector machine (SVM), or logistic regression (LR), for training. The output model can be used for the authentication process [55, 70]. All of the conducted methods followed the same procedure with different features or different algorithms.

These systems have shown a high accuracy in authenticating users. However, since the accuracy is not 100%, it may not authenticate the real user of the smartphone. To overcome this issue, these systems are used as a second authentication system and reauthentication [57] or continuous authentication systems [71, 72]. In this method, username and password are still utilized for authentication; however, for continuous authentication of the user during sessions, the phone keeps track of the user's touching and tapping behavior.

4.1.2. Personal Traits. Predicting personal traits from smartphone usage has been covered over the past decade. The conducted works started with surveys and questionnaires filled by the smartphone owners to obtain more insights into their

psychological traits. In [73], five different traits, named, the big five, have been studied. These characteristics are agreeableness, conscientiousness, extraversion, neuroticism, and openness. Logistic regression and linear regression have been leveraged to analyze the questionnaire results. Phone calls, texting, browsing, and gaming have been studied with age, sex, and genders. The authors claim a positive relation of agreeableness and phone calls but a negative relation with short messages. This relation has also been reported in [74] which means “less agreeableness more phone usage.” In addition, they reported that more gaming means less agreeableness [75].

Other personal traits have been conducted to assess the interaction between elderly people and smartphones [76]. Questionnaires and smartphone using patterns have been recorded for three smartphone apps. They attempted to study the relation between age and screen touching patterns. The results may be utilized for app designing enhancement or predicting users' ages. In [77], more than 13 ML classifiers have been compared to distinguish children from adults leveraging their keyboard touching. With more than 92% accuracy, the system has future potentials.

Another interesting example characterizing smartphone users' conditions through the touchscreen is the decline in the fine motor skills of smartphone users in cold weathers when their finger temperatures drop [78, 79]. This condition can be employed to study users' locations, health conditions, or other issues. In [80], the authors obtained an accuracy that exceeds 90% in gender classification. However, in [81], they reported an accuracy of only 61%. Even though the number of sensors used in [81] exceeds the number in [80], the selected ML algorithm and features have been optimized.

Another example in this category is the work conducted in [82] in which the touchscreen users' gaming behaviors have been collected and leveraged to predict the users who are playing. This method has recorded an accuracy of 80%. Moreover, the proposed Falcon app utilizes users' behaviors to reduce apps' start-up time [83].

Another interesting work was conducted in [84], where the authors attempted to detect the mood of the smartphone users by leveraging the data harvested from the sensors. The users' mood is not counted as personal traits; however, in the future, people may be personalized through their moods.

Other personal traits that have been harvested from smartphones are physical traits, such as sex, weight, height, age, race, and even shoe size. These traits have been estimated utilizing different smartphone sensors. Predicting these traits from smartphone sensors is called soft biometrics. In [85], a survey of a massive number of soft biometrics and their applications. Moreover, in [86], the challenges and the opportunities in this field were shown. It is worth mentioning that accelerometer sensor features have dominated in this area. Nevertheless, a fingerprint sensor utilized in a smartphone has also been utilized for gender and age classification [87–89]. Table 3 summarizes some of the interesting and early works conducted in this field. One thing to be mentioned is that personal traits estimated from smartphone sensors are affected by clothes and shoes [90, 91].

TABLE 3: Personal trait prediction.

Work	Sensor	Features	Algorithm	Traits	Results
[186]	Accelerometer	Time-domain features	ANN, J48 decision tree algorithms [187], and instance-based learning (IBk) [188]	Weight, height, and gender	71.2% for gender using IBk, 85.7% for height using ANN, and 78.9% for weight using IBk
[182]	Accelerometer and touchscreen	Time-domain features, touch pressure, and size	K -mean nearest neighbor	User identification	More than 96% for identification
[77]	Touchscreen	Delay between pressing two different keys	ANN, nearest neighbor, SVM, gradient descent bp, Euclidean distance, linear discriminant analysis, and another 5 algorithms	Classifying children from adults	More than 92% for SVM and 89% for linear discriminant analysis
[80]	Touchscreen	Delay and duration of pressing	SVM	Gender classification	Accuracy of 91%
[81]	Touchscreen, accelerometer, and gyroscope	29 features including: special keys, total keys pressed, number of backspaces used, edit distance, total completion time, average time between keys	Decision tree (number of keys), SVC linear kernel (age), SVC linear kernel (gender), logistic regression, K -nearest, and Gaussian NB	Number of fingers used, gender, and age	80% for the number of fingers, 75% for age, and 60% for gender
[189]	Touchscreen gestures, gyroscope, accelerometer	14 gesture features, total length, total time, width, height, area, pressure, speed, acceleration, arc distance, and angle start to end	SVM, logistic regression, naive Bayes, J48	Gender classification	71% accuracy for logistic regression
[190]	Fingerprint	Wavelet features and singular value decomposition	K -nearest	Gender classification	Accuracy exceeded 88%
[87]	Touchscreen	Swipe gesture speed in four directions and other features from [189]	Statistical	Thumb length and users' height	Accuracy of 72% of the relation between thumb length and height

4.1.3. Device Fingerprint. Device fingerprinting is defined as a method to detect and distinguish different smart devices even if they were manufactured by the same company in the same day at the same location. Researchers found that the electronic sensors designed and implemented in smart devices have certain noisy outputs that can be leveraged as fingerprints of these devices. Table 4 summarizes some of the works that have been conducted to distinguish smart devices according to their fingerprints.

4.1.4. Users' Status. Users' status is divided into two categories: activity and indoor localization. An overview of these two categories is presented below.

(1) Users' Activity. This is a massive umbrella that covers multiactions. Nevertheless, these actions can be divided into three main classes: simple, complex, and healthy. Simple activities can be defined with one action, such as walking, going upstairs, going downstairs, laying down, and sleeping. Complex activities combine different actions that happen at the same time, such as driving a car, riding a bicycle, or changing clothes. Finally, healthy activities are complex activities that combine multiactions that impact the health of the users, such as exercising and falling. Motion sensors have dominated in these applications. They have a higher accuracy than other sensors. However, what is the motivation of detecting users' status? To answer this question, few examples of activity detections will be shown.

In [92], the authors claimed that extracting users' activities from accelerometer and gyroscope data predicts the behavior of a car driver. They classified drivers into aggressive and normal drivers. The DTW algorithm has been implemented with time-domain features. The authors reported that the gyroscope data enhances the accuracy of the accelerometer data in predicting drivers' behaviors. In [93], the authors utilized an accelerometer and a gyroscope in detecting drunk drivers also. The authors proposed and designed an app that detects if the driver is drunk, alerts the driver, and calls the police. A statistical algorithm has been deployed in real time.

In [94], the authors attempted to classify the transportation methods (walking, biking, car, bus, and rail) exploiting GPS and accelerometer data in real time. They attempted to reduce the feature vector as much as possible to reduce the computational power. KNN and random forest have been compared for the classification purpose. Principal component analysis (PCA) and recursive feature elimination (RFE) have been used for the feature reduction process. An accuracy of more than 96% has been reported based on the random forest classifier. The reported data in this work can be leveraged to write a statistical report of transportation methods in cities. However, the conducted work requires GPS data. This sensor requires users' permissions to operate and harvest data. Physical activity detection is another application of users' status prediction. In this class, smartphones are used to classify the physical activities, such as walking, riding a bicycle, or sleeping. In [95], the authors utilized accelerometer and gyroscope readings with the SVM

algorithm to classify physical activities. Six different activities have been classified. 17 different time-domain and frequency-domain features have been extracted from these sensors. The authors reported an accuracy of more than 95% for walking, 79% for going downstairs, 72% for going upstairs, 92% for standing, 94% for sitting, and 100% for laying down. In [96], the authors compared deep-learning ANN with multiple algorithms based on the same motion sensors. They reported that deep-learning ANN exceeded a 95% accuracy compared with other algorithms. Nevertheless, they reported that SVM has a higher accuracy for stationary activities. In [97], the authors attempted to measure the performance of 6 different positions of smartphones with the users. SVM, KNN, and random forest algorithms with time, frequency, and wavelet features are used to compare the performance of the accuracy of different users' activities according to smartphone positions. In [98, 99], physical activity detection has been employed based on a magnetometer sensor to reduce the noise of the accelerometer sensor specially when locating the smartphone in different body areas. In [97], more than 27k data samples have been harvested from ten different subjects. Wavelet, frequency, and time-domain features have been extracted from multimotion sensors. Multialgorithms have been utilized, such as random forest, SVM, and KNN. The extracted results show a high accuracy in daily activity prediction. In [100], Actitracker has been proposed to exploit the harvested data from motion sensors to detect users' physical activities as a health monitoring application. Time- and frequency-domain features have been collected and fed to a random forest classifier. The Actitracker application allows the users to define a threshold for their daily activities to measure them. In [101], Happito, an activity tracker smartphone app, has been assessed. The study shows that the users access the app for 5 seconds on average to check their status only and they have no interest in their historical logs. This shows that these data should be deleted in daily bases for security purposes.

In the health monitoring field, different diseases have been detected using motion sensors, such as Parkinson's disease, epilepsy, and strokes [102, 103]. Fall detection applications dominated in this area. In [104], the authors utilized four different classification algorithms: naive Bayes, J48 decision tree, random forest decision tree, and SVM, to detect a fall. Four types of falling have been recorded: forward using hands, forward using knees, sideward, and backward. The authors claim that the accuracy exceeds 99% for all time and frequency features. Other works utilized other machine learning algorithms for fall detection with time, frequency, and wavelet features [105, 106]. Finally, an application has been written utilizing motion sensor data for fall detection and alarm [107].

A smartwatch has been leveraged in this field. In [108], the authors utilized a smartwatch to recognize six different activities utilizing five different MLAs. An accuracy of over 90% has been recorded for detecting drunk people. In [109], six different activities with three different algorithms using time-domain features have been proposed. A 90% accuracy has been reported for the J48 algorithm. An interesting physical action detection is the step count application.

TABLE 4: Device fingerprint classification.

Work	Devices	Sensors	Scenario	Features	Algorithms	Results
[18]	20 Androids	Gyroscope, accelerometer, magnetometer, microphone, and vibrator	4 scenarios: (a) smartphone on a table with and without vibration and (a) smartphone held in the hand with and without vibration	Time-domain and frequency-domain features	Random forest and naive Bayes	Accelerometer accuracy higher than both sensors. With the combination of all sensors, the identification accuracy exceeds 90%
[191]	17 Androids and 17 IOS	Microphone, speakers, and accelerometer	Three scenarios (wooden desk, metal cabinet, and windowsill)	Frequency response and FFT value	Maximum likelihood estimation (MLE), simple Euclidean distance-based classification, and K-NN classification	95% accuracy with a microphone and speaker and more than 98% for both
[180]	10 Androids	Accelerometer, gyroscope, magnetometer, microphone, camera, and vibrator	Flat wooden surface and hand-held	Photo response nonuniformity (PRNU), time-domain and frequency-domain features	Bagged decision tree	High accuracy for gyroscope and accelerometer, 100% for the combination of both
[192]	4 IOS, 1 Blackberry, and 8 Androids	Camera	—	Wavelet features, photo response nonuniformity (PRNU)	SVM	Accuracy of approximately 94%
[193]	8000 IOS	All sensors and context features	—	29 different features	SVM and random classifier	Accuracy of approximately 97%
[194]	6 cameras and 3 smartphones	Camera	—	Color, quality, and frequency-domain features	SVM	Accuracy between 66% and 97%
[195]	12 smartphones and camera	Camera	—	Color, quality, frequency domain, and wavelet feature +PRUN	SVM	For all features, accuracy increases. Some features obtain better results in specific scenarios
[196]	Arduino and accelerometer	Accelerometer	On a flat table	Time-domain features	Statistical	Each accelerometer chip has its own fingerprint
[25]	3 smartphones from three vendors	Accelerometer and gyroscope	On a flat table	Time-domain features	SVM	Accuracy more than 90%
[181]	30 between IOS and Android	Accelerometer and gyroscope	On a table	Time- and frequency-domain features	SVM, naive Bayes, multiclass decision tree, K-nearest neighbor (KNN), quadratic discriminant Analysis (QDA) classifier and Bagged Decision Trees	Bagged decision trees have the highest accuracy

In this application, the classifier first detects the footsteps; subsequently, it attempts to count these steps [110]. This application can be used as a first step in an indoor localization process.

(2) *Indoor Localization.* GPS has dominated in the outdoor localization system. A GPS receiver is embedded in all new smart devices. However, an indoor environment is a GPS-free domain. This made it a hot research topic in the last few years especially in the localization process in subways,

skyscrapers, and malls. Indoor localization is divided mainly into two classes, Wi-Fi localization and pedestrian dead reckoning (PDR). The first class has a high accuracy. However, it requires installation of network infrastructure and access points. In [111], smartphone sensors and Wi-Fi signal have been utilized to construct an accurate indoor localization system with an error rate of approximately 1.1 m. The KNN algorithm has been adopted for its simplicity. To eliminate access point installation and the infrastructure, PDR is proposed. In [112], gyroscope and accelerometer data have been

recorded for indoor localization. In [113], authors proposed an accurate indoor localization and tracking app based on a magnetometer and camera. A neural network model has been written for image comparison. Three different facts have been claimed in this study. First, the magnetic reading of the sensors differs according to its location in the building. Second, the reading does not depend on time. Finally, the magnetic reading is semi-immune to background noise. Other studies have shown that magnetometers, along with accelerometers, can be used to build tracking systems with very high accuracies that can work indoors effectively and have low power consumption unlike GPS [114] in PDR apps. Other researchers have shown a method to create indoor maps for buildings using magnetometers and accelerometers [115]. Another type of indoor localization is altitude localization. In this field, a barometer sensor dominated. A barometer can build models of detecting the location of the phone, and hence the phone user, inside buildings with 100% accuracy [116]. Phone barometers were also utilized to detect the floor level of the user with high accuracy [40]. Other studies have shown that a user's location can be estimated and tracked with decent accuracy by only using the phone barometer [117]. Although accelerometers can also be used in indoor localization techniques, barometers were proven to be more accurate especially when the phone is distracted with other activities like gaming or a phone call [116]. Another study demonstrated accurate readings of the phone altitude primarily by using the phone gyroscope along with the accelerometer [118].

4.1.5. Health Applications. Smart devices in health applications have proliferated in the past decade. These applications are classified into three main domains: real-time health monitoring, health activity tracking, and health issue and disease detections. In health monitoring applications, smart devices can be leveraged to monitor different aspects and parts of the human body in active or passive modes. In the active mode, the user is responsible for performing a certain operation utilizing the smart device to read the internal organic signals. For examples, in the Cardiio app [119, 120], the smartphone camera has been adopted to measure the heartbeat through detecting the changes in the skin color while the blood is circulating through the body. In contactless health monitoring applications, acoustic signals have dominated. Microphones and speakers have been heavily adopted. In [121], a smartphone app has been proposed to utilize the microphone and the speaker to monitor the heartbeat. In [19], acoustic signal-based sleep quality monitoring application has been proposed. Another health monitoring application is the rehabilitation process monitoring after injury. In [122], gyroscope and accelerometer data have been recorded at home to track the rehabilitation progress of total knee arthroplasty.

In health activity tracking, smart devices and their sensor-based apps are proliferated in the literature. In [123], a stroke tracking and preventing smartphone app has been designed.

In disease detection, smart devices have shown a massive potential. Different smart device sensors have been leveraged in different applications. In [124], a smartphone camera has

been used for blood hemoglobin testing for anemia. Accuracies between 76% and 85% have been recorded. Another example of contact or active camera testing applications has been proposed in [125] to test the skin lesion for different bacterial diseases, such as Buruli ulcer. In [126], all smartphones' sensor data have been recorded to monitor mental health and detect depression, stress, and loneliness. Another example of disease detection as a smartphone app is the detection of the impact of skin diseases on the fingerprint identification process. It has been found that some skin disease symptoms may affect the skin color or the structure of the papillary ridges which may affect the fingerprint scanners [127, 128]. Moreover, studies have shown that there is a correlation between the fingerprint patterns and diabetes. Kahn et al. found that diabetes was associated with the mean dermatoglyphic ridge count difference between the thumb and the little finger given the adjustments of gender and age [129]. Others also showed that the fingerprint whorls, loops, and arches in diabetic patients significantly differ from non-diabetics [130, 131]. Although fingerprint patterns are only associated with the diagnosis of genetic-based diseases [132], this still raises the question of whether mobile phones, equipped with fingerprint scanners, will be able to perform such tasks like predicting the development of diabetes or detecting certain types of skin disease in users.

4.2. Accuracy of the Extracted Information. As mentioned in the previous section, many useful applications have been proposed and developed based on the training process of different datasets. The training process and testing and validation of these applications have been conducted in a controlled environment. Moreover, the harvested data is filtered before being utilized in MLA. These issues raised questions about the accuracy of the developed applications in real life and outside the controlled environment [133].

Another revealed problem is the number of features that have been extracted and utilized in different applications. As mentioned in the previous section, the same features have been utilized over and over again to extract different conclusions. The same features have been utilized for personal traits and for personal activities. If the same features reveal all of the information, how will the personal traits not impact the extracted activities? For example, in [134], the authors show how noisy data in big data smartphone health applications may lead to misleading conclusion. The authors studied the accuracy of a step count app in Apple and Android smartphones. They revealed a large error range in these apps in both platforms. In [135], a study of the quality of experience of smartphone health apps has been conducted. The obtained results revealed different questions from the users on the validity, the accuracy, and the privacy of the information. This shows that accuracy is one of users' concerns. In [136], the authors compared the classification of smartphone apps of personal daily activities of two different groups, the first group of 20 young people and the second group of 37 old people. They trained the classifier from the data harvested from the first group and tested the module on the second group. The same experiment has been repeated with another module trained from the second group data and tested on the

TABLE 5: Side-channel attacks.

Work	Type	Sensors	Features	Comments	Results	Algorithms
[58]	Motion	Gyroscope and touchscreen	Time domain, angle of upper bisector, and angle of lower bisector	Digit-only soft keyboard	70% accuracy for a 4-digit PIN	Guess classifier
TapLogger [197]	Motion	Gyroscope, accelerometer, and touchscreen	Time domain and angle changes	Soft keyboard for digits	90% accuracy with 3 traits for an 8-digit PIN	SVM using LIBSVM [198]
TapPrints [183]	Motion	Gyroscope and Accelerometer	Time domain, frequency domain, and FFT values	Soft keyboard for English characters	90% accuracy for English character inferring	K -nearest neighbor (KNN), multinomial logistic regression, SVM, random forests [199], and bagged decision trees
Accessory [184]	Motion	Accelerometer	Time domain, the average time from a sample to a peak, the total time of the window, and the number of samples in the window	Soft keyboard for English characters	6 password characters in 4.5 trails	Random forest, ANN, SVM, and C4.5 decision tree
[200]	Acoustic	Microphone	Cepstrum features [201] and speech recognition, claim that it is better than FFT	Soft keyboard for English characters	96% accuracy	Hidden Markov models [202], linear classification, ANN, and language models have been used
Timing attack [142]	Timing	Microphone	FFT	Soft keyboard and hard keypad	Inferring PIN without triggering alerts	Markov chains with brute force attack
Soundminer [203]	Acoustic	Microphone	FFT, voice record, and dual-tone multifrequency (DTMF)	Soft keyboard	Inferring PIN, passwords, volume up and down keys	Google speech recognition
Powerspy [145]	Power	Power usage of wireless communication	Power usage files as time series	—	Inferring routes and real-time tracking dynamic	Time warping (DTW) [204] and optimal subsequence bijection (OSB) [205]
[149]	Power	Power usage of the smartphone	Power usage files as time series	—	Inferring apps, geolocation, password length, and UI	Statistics
[153]	Timing	Interrupts	Interrupt file	—	Inferring apps and unlocking the phone	DWT and hidden Markov model (HMM)
PinMe [151]	Motion	Accelerometer, gyroscope, barometer, IP, and time zone	Sensor data	—	Inferring and tracking users around the world	SVM
[157]	Acoustic	Microphone	Row data	—	Accuracy of more than 70% for 5-minute audio files	Statistical
[156]	Acoustic	Microphone	Row data	—	1.5 error rate	Unsupervised algorithm

first group. They reported a massive impact on the accuracy in both scenarios. This means that personal traits should be added as features in these studies or a massive dataset should be harvested from different countries with all ages. In [137], the authors conducted a gait recognition experiment utilizing a smartwatch gyroscope and accelerometer using data from the same day and different days for the testing and the validation process. They reported an increase in the reported errors for the gait recognition when data on different days are utilized.

A third problem is the size of the dataset. If these apps will be used in different countries and from people with different ages, how should the dataset be harvested and how big should it be? In [138], the authors conducted an experiment with the data harvested from more than 700k people from 111 countries to study countries' obesity situations. Other studies utilized 10-50 participants only [134, 135].

The accuracy question of smart device apps utilizing MLA requires different procedures of testing and validation in real life.

5. Hidden Information Inferring Issues

As mentioned above, many useful applications have been proposed through utilizing sensor open-access hidden information. However, many issues are revealed exploiting these data. In the following sections, security attacks and privacy issues are introduced. Moreover, other real physical security issues will be revealed.

5.1. Side-Channel Attacks. Side-channel attacks are defined as any kind of computer attacks that can be implemented exploiting harvested data from a system in legal ways rather than bugs in the deployed algorithms [139]. These attacks are divided into nine main categories. Four of these categories have been implemented in smartphones as shown in Table 5.

The idea of a side-channel attack started long time ago. Many methods and algorithms have emerged in this area. One of the oldest methods is electromagnetic emanation that has been proposed in the 1980s. In this method, researchers discovered that electronic components emit electromagnetic waves while switching between different states. This method has been exploited in different attacks. In [140, 141], the authors attempted to infer computer passwords through keystrokes' electromagnetic waves. The authors claimed that pressing a key will emit electromagnetic waves that can infer the pressed key. Different scenarios have been assessed, such as falling edge and rising edge. In [142], the authors attempted to study the unique audio feedback when pressing keys to infer the pressed keys. Moreover, the authors attempted to study the distance from the keyboard to recognize these feedback sounds. They found that this method can be exploited with brute force attack without triggering any alerts.

The side-channel attacks have moved to smartphones by exploiting their sensors. Some sensor data is harder to obtain compared with others. For example, GPS data requires

smartphone users' permissions to start the harvesting process. However, other sensors do not require permissions. For example, W3C published DeviceOrientation Event Specification which allows JavaScript in websites to access accelerometer and gyroscope data in Android and IOS without the user's permissions [143]. One of the first works that have been conducted to reveal sensor data attack threats and defend sensor attacks is found in [144]. Many works have been conducted to show that all types of side-channel attacks are viable in smartphones. For example, power usage as a side-channel attack has been exploited in [145]. Two open-access files are present in Android smartphones that track power usage (`/sys/class/power supply/battery/voltage now` and `/sys/class/power supply/battery/current now`). Any process or app can access these files without permission. The authors utilized these files for smartphone tracking and route distinguishability. Two scenarios have been shown for distinguishing routes, real-time tracking, and inferring new routes. Two machine learning algorithms have been leveraged. The authors showed how power usage of a smartphone is the same for the same route even if two different smartphones are used. The proposed method does not require cell IDs nor access points SSID such as in [146–148]. All of these methods trace smartphones and infer routes based on power side-channel attacks. Another example of power usage as a side attack has been shown in [149]. This work utilized the same two power monitoring files. Four different attacks have been shown: app identification, UI inferring, password length inferring, and geolocation. They have shown how a statistical method is used to obtain accurate results in all of these situations. In [150], the authors have shown how the power traces can be used to distinguish between different cryptographically algorithms in Android smartphones.

Another example of side-channel attacks is motion attacks. In [151], the authors proposed PinMe, an algorithm that can track users around the world. Time zone, IP address, and accelerometer, gyroscope, and barometer sensors have been leveraged. PinMe can trace users while performing different actions, such as walking, driving, and being on a train and even on a plane. In [152], the authors proposed a method that leverages the accelerometer and gyroscope to find routes in a city that a user drives in. A search algorithm based on a map as a graph has been proposed. The method has been tested on 30 cities with an accuracy of more than 50% to find a list of ten possible routes.

Another example of side-channel attacks is timing attacks. In [153], the authors showed another open-access files in an Android platform, called (`/proc/interrupts`). This file keeps track of all hardware interrupt requests in the system. Utilizing this file, the authors successfully inferred the lock patterns, distinguished UIs, and identified apps.

In [154], the authors showed how public open-access zero permission controlling and monitoring files are exploited for different attacks. For example, the file (`/proc/uid-stat/`), which shows statistics of the application network usages, can be shown to infer the installed applications and most popular apps the users use. A case has been shown to infer the health condition of the user by inferring diseases 'articles' pagesthat users read in the WebMD app. In addition, the

authors have shown how the file (/proc/net/arp) may be utilized to infer users' locations.

In [155], a light side-channel attack has been proposed based on the data harvested from an ambient light sensor. The author has shown that light intensity recorded by this sensor changes with finger tapping position on the touchscreen. This process has been exploited to predict the PIN code entered using tapping on the smartphone touchscreen. The author has shown a high accuracy in detecting PIN codes. However, other techniques should be included to enhance the accuracy.

Another interesting side-channel attack is proposed in [156]. The authors claimed that the number of people in allocation can be counted utilizing a microphone and unsupervised MLA. An error of 1.5 has been recorded with different levels of noise in the background.

Finally, an interesting new acoustic side-channel attack has been shown in [157]. In this method, the authors claimed that any recorded voice or video files around the world have a location fingerprint. This fingerprint comes from electrical network frequency (ENF) signals that can be detected in the recorded files. By downloading hundreds of videos from YouTube from different countries and cities and then extracting ENF information of these videos, the new recorded sound or video files can be compared with the downloaded files to find similar ENF. The authors claimed an accuracy of over 70% for audio files longer than 5 minutes. Another acoustic side attack has been shown in [158].

Table 5 shows a summary of side-channel attacks implemented in smartphones.

5.2. Privacy. Privacy is defined as the state of having no public attention. In other words, anyone keeps his/her life private without letting others know the details of his/her life. Nowadays, Internet and social networks allowed subscribers to contribute and share their photos, comments, locations, and their statuses. However, privacy has been considered in a new method that people or subscribers have control over what they share and whom to share with. This tuned the privacy definition to control of the content and people attention. With the big data era, MLAs allowed developers to interpret massive data in different ways from smart devices [159, 160]. This adds burden in the designing of algorithms and the type of harvested data. In [8], the author showed that one of the biggest privacy issues in IoE is that the users have limited control over what data to share and distribute. It has been mentioned in the human rights that humans have the rights to keep private things secret [161]. However, as mentioned earlier, big data may interfere with this right through digging secret information from freely available meaningless data.

The main problem in privacy in sensed data mining is that it is implicit. Users do not know what information can be detected from their own sensors. In the questionnaires that have been conducted in [162], the authors attempted to measure the confidentiality of smart device users. The study compared the confidentiality level of computer users and smartphone users. Nine different categorized questions have been written in the questionnaire. The authors' study

showed that 68% of smartphone subscribers would not enter their PIN code in smartphones for privacy and security concerns. Moreover, for the health reports, 38% of the people are willing not to open such reports utilizing their smartphones. Nine people have said "The more health problems you have, the more potentially private they become, and the more private they become, the less likely I am to do it on a cell phone." Finally, for the location services, most of the people who are afraid of leveraging such a service comment that they are afraid of robbery. This survey shows that smart device subscribers have concerns for their privacy. However, what can they do to keep their privacy from data mining?

In [163], the authors have shown that smart device privacy is complicated since it consists of different layers of hardware, operating system, and apps. Another layer of the sensor data mining process is added over this layered stack. This shows how the privacy of smart devices in the era of IoE requires a new arrangement to enhance it without any impact on the usability of the devices.

5.3. Security Threat Scenarios. In this section, some of the real-life attacks of smart devices will be shown. These attacks sometimes can be categorized as part of side-channel attacks. However, these attacks exploit some physical phenomena utilizing smart devices. For example, in [158], the authors exploits the microphone frequency sensitivity levels to modulate a command that is inaudible for humans. The command can be collected from a smartphone microphone and interpreted and start a sequence of actions utilizing the voice assistant service, such as Siri. A command is recorded and modulated with a signal of frequency higher than 20 kHz. The authors claimed a success with a very high accuracy. In [164], the authors utilized the accelerometer and gyroscope in a smartwatch to detect the mechanical lock combinations. In other words, whenever the smartwatch users open a safe, the code can be detected by any app harvesting the data of the accelerometer and gyroscope. This attack may be introduced as a side-channel attack; however, the harvested data has been utilized to hack real-world equipment. This is why we think it belongs to this category. Many other security threats can be exploited. Another example has been proposed in [165] to infer information of the manufacturing plane and machines exploiting the magnetometer and microphone. The authors succeeded in distinguishing CNC machines, 3D printer, and their types and kinds. In the following, we show three different potential scenarios that can be implemented in the future.

The first scenario is house burglary. In this scenario, the burglar requires three pieces of information to successfully rob a house: owners' activity, location, and number of people in the house. In [19, 20], the authors have shown how the sleeping pattern of smartphone users can be recorded utilizing touchscreen light, battery charging status, and cable plugged to the charger or not. These features can be mined to study the house owner's activity combined with the accelerometer and gyroscope [160]. The second piece of information is the location in the house. In Indoor Localization, we found that it is possible to locate a user in a closed area utilizing only the accelerometer and gyroscope. To count the people in the house, microphone data can be used as in [156].

The second scenario is the location tracking as in [166]. If users' touchscreen behaviors have been harvested and recorded from different smartphone apps over a long time period. This data can be indirectly exploited to track and find a user. Even if the user changed the smartphone device and create new mail accounts and new names and passwords, apps and companies can track users to show the similarity between any new users with the existing users. In other words, no one will start from scratch again in this digital world. Moreover, if your touching behaviors are harvested for a long time, heat maps can be generated to visualize this data before analysis. Subsequently, these data can be utilized to predict users' credentials, such as usernames and passwords for different applications. In other words, a new era of key logger software can be written.

The third scenario is personal trait-driven attacks. Assume a virus that only hacks the smart devices of women or children. Such viruses can be distributed over different devices; however, it works only according to the users' behaviors. Moreover, an app is triggered to start/shutdown according to the users' mood.

This will open the door of new behavior/personal trait attacks.

These are few examples of thousands of security threats and scenarios that can be implemented and proposed. These scenarios are an open field for innovation.

5.4. How and Where to Start Digging Hidden Information.

With the proliferating phenomenon of smartphones, tablets, smart TVs, and smartwatches, there has been a rapid development in the tools and methods of developing apps for them. Many methods and tools have been proposed, designed, and commercialized [167]. Nevertheless, these tools are fitted into one of three classes: programmable tools, zero-line coding tools, and hybrid tools. The programming tools are defined as the integrated development environments (IDEs) that require skills in at least one programming language to program a decent app. Android studio [168], which requires deep skills in XML and Java programming, is the official IDE for Android app development. DroidEdit [169] is another example of this category that also requires skills in Java programming language. A third example is Cordova, which requires skills in web front end programming languages, such as CSS, HTML, and JavaScript. This category of tools, as mentioned, requires deep knowledge in computer science and programming languages to develop apps. In this case, it will be easier to track and trace the code back to its designers [170]. Moreover, it is hard for amateur hackers or crackers to write apps to harvest users' sensor data, update them to a server, and keep the application usage as low as possible. However, this programming style may reduce the usability of smart devices. In [171], the authors attempted to utilize Android studio to write an app to harvest users' sensor data and upload them to a server. The authors attempted to show legal and ethics of utilizing such data.

Another group of tools has been proposed: zero-line coding tools [172]. In this category, an application is utilized to convert web applications and web pages into smart device apps. Any online web application or site can be converted

into an app without writing any single line of code. This tool is dangerous since some smart device sensors can be accessed from JavaScript without any permissions [143] as mentioned in Section 5.1. However, the app designers should first have a web application to be converted. This also requires deep skills.

The third and the most sophisticated smart device app development tool is the hybrid one [173]. In this tool, the simple logical flow of the app is required to design a complex code. No programming skills are required. However, algorithm writing is required. One of the most popular examples of this tool is MIT App Inventor (MAI) [174], which is defined as an event-driven programming style. MAI allows programmers to obtain all the functionalities of any complex IDE without any programming skills. Any sensor can be harvested. Data can be exchanged in an easy way utilizing Wi-Fi, mobile network, Bluetooth, and NFC [175]. A Google free account is the only requirement to start writing any complex app. This environment has been utilized in a smart home monitoring app [176], a fitness app [177], a health monitoring app [178], and smart lamp design utilizing smartphone sensors [179].

The hybrid tool category shows that side-channel attacks are easy to implement. To start implementing apps, Figure 9 shows the required steps. Firstly, data dumps from smart devices are required. These dumps as shown are available over the web. Any of these dumps can be downloaded. Secondly, a feature extraction process should be implemented. As mentioned, time-, frequency-, and wavelet-domain and row data features can be extracted from the data. Finally, a machine learning engine, such as R, Python, or MATLAB, is required to compare different MLAs to adopt the accurate and simple one. Sometimes adopting the easiest one in implementation dominated over the accurate one. Finally, the mathematical model is ready for deployment.

To deploy the trained algorithm in the real world, a smart device app is required. MAI simplifies this task. The algorithm should be embedded in any type of apps. The deployment steps are shown in Figure 10. The deployment process consists of two main parts: client side and server side. The client side is the smart device app. This app should at least contain four different modules. The first module is the timer module which will record sensor reading over preconfigured periods. Moreover, the time stamp of sensor data harvesting has been utilized as a feature in different algorithms as was shown. The second module is sensor modules. What type of sensor data has been utilized in the training process should be harvested in this step. All sensors, except for the fingerprint sensor, are implemented in MAI. The third module is the data saving module. This module is required to reduce network usage, and any data processing modules required in the smart device app. MAI allows the programmer to save the app data in an internal unique database. The final step is data transmission over the Internet module. The HTTP protocol can be adopted for this step.

In the server side, a web application should be written and hosted over the web. The application should extract any received data from the data transmission process. The Internet protocol (IP) address of the sender should be

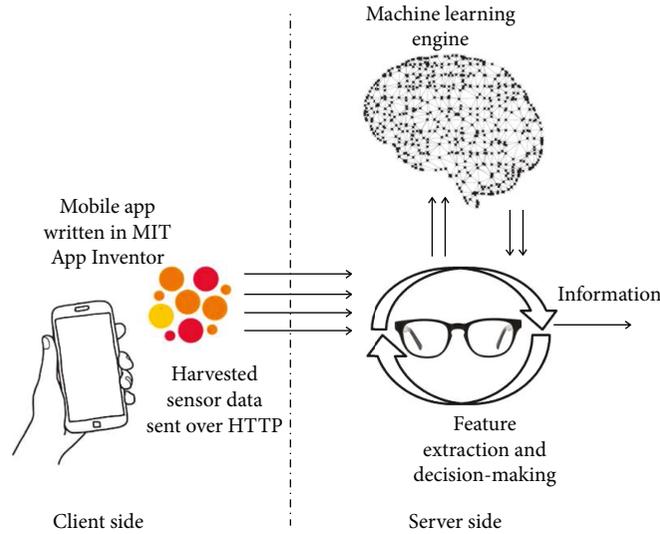


FIGURE 9: The training module of MLA.

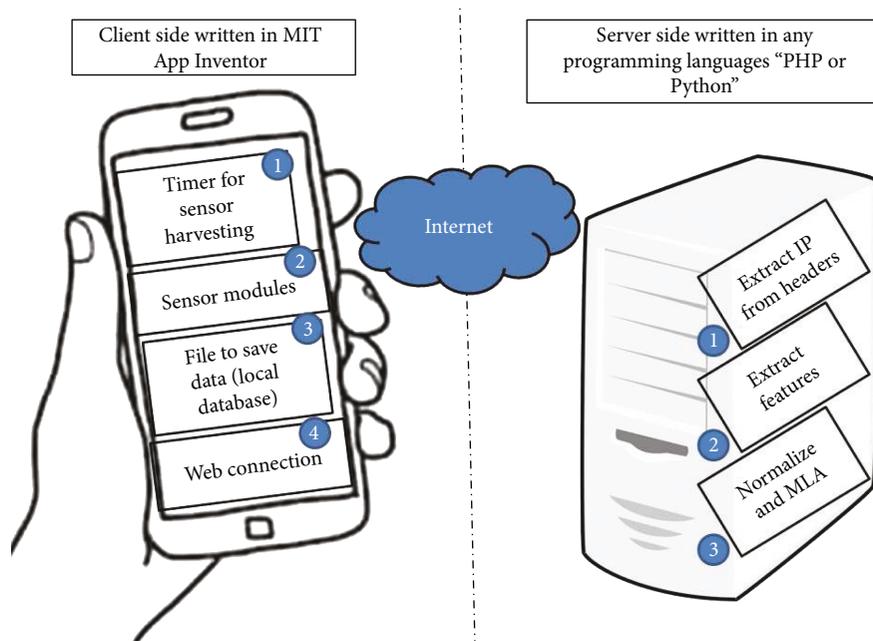


FIGURE 10: The implementation process.

recorded to distinguish app users. Moreover, it should record time stamps. The second step in this application is feature extraction from the received data. Finally, the trained mathematical module is employed in the harvested features to obtain the hidden information. Other choices can be made.

Another method may be utilized to reduce network usage. All the steps are moved to the smart device app. In this method, the network usage will be reduced to the minimum since the device will only send the hidden information. However, the computational load will increase. To reduce the computational load of the app, the timer module

can be configured to employ the feature extraction and the MLA mathematical module in very long periods. These two implementation scenarios show how easy it is to breach the security of smart device users in the IoE era.

6. Conclusion and Discussion

Smart devices are everywhere. The IoE era has arrived. The advantages, applications, and usability of this paradigm have been introduced in many research papers. The privacy and the security of smart devices in IoE have attracted researchers over the years to construct secure systems. Nevertheless,

machine learning and big data complicated the story. In this paper, we show how machine learning, big data, and smart devices' sensor data are exploited to find many useful hidden information. It has been shown how smart device sensors, which are utilized to enhance the usability of the devices, may be leveraged in useful applications on the one hand and in hacking and attacking issues on the other hand. Moreover, it has been shown how these threats and attacks can be implemented and deployed in a simple method utilizing event-driven programming without deep programming skills.

Unfortunately, there is no hidden data protection manual that can be downloaded and followed to solve the accuracy, privacy, and security issues. However, many techniques can be utilized from app designers and users to reduce these issues as much as possible.

User awareness is the most important step to prevent hidden data issues. Users should be aware of what to upload to the Internet. App permissions should be read carefully before installing new apps. Users should not install apps from unknown sources or developers. Users should not grant any permission required from any app until they think why such an app requires such permissions. For example, different games on the Android market require access to the smartphone media and files, why? Users should be smarter than their smart devices.

Smart device operating system developers should increase and enhance the permissions on smart device sensor access. More control should be granted to the users. More warning messages should be shown all the time. Do not show this again message should not be used. More research and development in this field are required. For the accuracy enhancement, more data should be harvested from different ages, genders, and countries to reduce the impact of different variables on the concluded output. The developed applications should be tested in real life through different users for a period of time before announcing the validity of its conclusions. Social networks are a fertilized environment for this step.

Finally, we believe that the static design of smart devices is one of the main issues in the area of hidden data threats. For example, many of the smart device users do not know what sensors they have and how to use them. Moreover, many sensors are useless for these users. If smart device users have the ability to design and configure their devices with only the necessary sensors and parts, a part of this issue will be solved.

Conflicts of Interest

The authors declare no conflicts of interest.

References

- [1] L. Atzori, A. Iera, and G. Morabito, "The Internet of things: a survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [2] L. Mainetti, L. Patrono, and A. Vilei, "Evolution of wireless sensor networks towards the Internet of things: a survey," in *SoftCOM 2011, 19th International Conference on Software, Telecommunications and Computer Networks*, pp. 1–6, Split, Croatia, 2011.
- [3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: a survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [4] L. Da Xu, W. He, and S. Li, "Internet of things in industries: a survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
- [5] E. Borgia, "The Internet of things vision: key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1–31, 2014.
- [6] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for Internet of things," *Journal of Network and Computer Applications*, vol. 42, pp. 120–134, 2014.
- [7] M. A. El Khaddar and M. Boulmalf, "Smartphone: the ultimate IoT and IoE device," in *Smartphones from an Applied Research Perspective*, INTECH, 2017.
- [8] M.-H. Maras, "Internet of things: security and privacy implications," *International Data Privacy Law*, vol. 5, no. 2, pp. 99–104, 2015.
- [9] D. Nunan and M. Di Domenico, "Big data: a normal accident waiting to happen?," *Journal of Business Ethics*, vol. 145, no. 3, pp. 481–491, 2017.
- [10] C. Perrow, *Normal Accidents: Living with High Risk Technologies-Updated Edition*, Princeton university press, 2011.
- [11] J.-E. Ekberg, K. Kostianen, and N. Asokan, "Trusted execution environments on mobile devices," in *CCS '13 Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1497–1498, Berlin, Germany, November 2013.
- [12] P. Benz, *Gesture-based interaction for games on multi-touch devices, [Ph.D. thesis]*, University of Cape Town, 2010.
- [13] F. Lettner and C. Holzmann, "Heat maps as a usability tool for multi-touch interaction in mobile applications," in *MUM '12 Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, pp. 49:1–49:2, Ulm, Germany, December 2012.
- [14] R.-D. Vatavu, L. Anthony, and J. O. Wobbrock, "Gesture heatmaps: understanding gesture performance with colorful visualizations," in *ICMI '14 Proceedings of the 16th International Conference on Multimodal Interaction*, pp. 172–179, Istanbul, Turkey, November 2014.
- [15] Appsee Company, "Appsee app," 2016, <https://www.appsee.com/>.
- [16] H. Luna-Garcia, A. Mendoza-Gonzalez, R. Mendoza-Gonzalez et al., "Analyzing typical mobile gestures in mHealth applications for users with Down syndrome," *Mobile Information Systems*, vol. 2018, Article ID 2830851, 9 pages, 2018.
- [17] Y. Rezik, R.-D. Vatavu, and L. Grisoni, "Understanding users' perceived difficulty of multi-touch gesture articulation," in *ICMI '14 Proceedings of the 16th International Conference on Multimodal Interaction*, pp. 232–239, Istanbul, Turkey, November 2014.
- [18] I. Amerini, R. Becarelli, R. Caldelli, A. Melani, and M. Niccolai, "Smartphone fingerprinting combining features of on-board sensors," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 10, pp. 2457–2466, 2017.

- [19] J.-K. Min, A. Doryab, J. Wiese, S. Amini, J. Zimmerman, and J. I. Hong, "Toss 'n' turn: smartphone as sleep and sleep quality detector," in *CHI '14 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 477–486, Toronto, Ontario, Canada, April–May 2014.
- [20] R. Nandakumar, S. Gollakota, and N. Watson, "Contactless sleep apnea detection on smartphones," in *MobiSys '15 Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, pp. 45–57, Florence, Italy, May 2015.
- [21] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Communications magazine*, vol. 48, no. 9, pp. 140–150, 2010.
- [22] M. Liu, "A study of mobile sensing using smartphones," *International Journal of Distributed Sensor Networks*, vol. 9, no. 3, Article ID 272916, 2013.
- [23] Y. Cai, Y. Zhao, X. Ding, and J. Fennelly, "Magnetometer basics for mobile phone applications," *Electronic Products*, vol. 54, no. 2, 2012.
- [24] T. Ozyagcilar, "Implementing a tilt-compensated eCompass using accelerometer and magnetometer sensors," *Freescale semiconductor*, vol. AN4248, 2012.
- [25] G. Baldini, G. Steri, F. Dimc, R. Giuliani, and R. Kamnik, "Experimental identification of smartphones using fingerprints of built-in micro-electro mechanical systems (MEMS)," *Sensors*, vol. 16, no. 6, article 818, 2016.
- [26] R. D. Labati, V. Piuri, and F. Scotti, *Touchless Fingerprint Biometrics*, CRC Press, 2015.
- [27] A. Pocovnicu, "Biometric security for cell phones," *Informatica Economică*, vol. 13, no. 1, 2009.
- [28] K. Delac and M. Grgic, "A survey of biometric recognition methods," in *46th International Symposium Electronics in Marine*, vol. 46, pp. 184–193, Zadar, Croatia, June 2004.
- [29] K. Cao and A. K. Jain, "Hacking mobile phones using 2D printed fingerprints," Technical Report, 2016, http://biometrics.cse.msu.edu/Publications/Fingerprint/CaoJain_HackingMobilePhonesUsing2DPrintedFingerprint_MSU-CSE-16-2.pdf.
- [30] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer Science & Business Media, 2009.
- [31] T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "'Impact of artificial' 'gummy' fingers on fingerprint systems," in *Optical Security and Counterfeit Deterrence Techniques IV*, vol. 4677, pp. 275–290, San Jose, CA, USA, April 2002.
- [32] S. Shigematsu, H. Morimura, Y. Tanabe, T. Adachi, and K. Machida, "A single-chip fingerprint sensor and identifier," *IEEE Journal of SolidState Circuits*, vol. 34, no. 12, pp. 1852–1859, 1999.
- [33] S. Jung, R. Thewes, T. Scheiter, K. F. Goser, and W. Weber, "A low-power and high-performance CMOS fingerprint sensing and encoding architecture," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 7, pp. 978–984, 1999.
- [34] L. Coetzee and E. C. Botha, "Fingerprint recognition in low quality images," *Pattern Recognition*, vol. 26, no. 10, pp. 1441–1460, 1993.
- [35] M. Tartagni and R. Guerrieri, "A fingerprint sensor based on the feedback capacitive sensing scheme," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 1, pp. 133–142, 1998.
- [36] T. Caldwell, "Voice and facial recognition will drive mobile finance," *Biometric Technology Today*, vol. 2012, no. 10, pp. 2–3, 2012.
- [37] J. Kanchikere and R. Sudha, "Hacking mobile phones using 2D printed fingerprint," *International Journal of Innovations & Advancement in Computer Science, IJIACS*, vol. 7, no. 4, 2018.
- [38] K. Cao and A. K. Jain, "Learning fingerprint reconstruction: from minutiae to image," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 104–117, 2015.
- [39] N. Ben-Asher, N. Kirschnick, H. Sieger, J. Meyer, A. Ben-Oved, and S. Moller, "On the need for different security methods on mobile phones," in *MobileHCI '11 Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, pp. 465–473, Stockholm, Sweden, August–September 2011.
- [40] H. Ye, T. Gu, X. Tao, and J. Lu, "Scalable floor localization using barometer on smartphone," *Wireless Communications and Mobile Computing*, vol. 16, no. 16, 2571 pages, 2016.
- [41] M. Wu, P. H. Pathak, and P. Mohapatra, "Monitoring building door events using barometer sensor in smartphones," in *UbiComp '15 Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 319–323, Osaka, Japan, September 2015.
- [42] M. R. Kamdar and M. J. Wu, "Prism: a data-driven platform for monitoring mental health," in *Biocomputing 2016*, pp. 333–344, Kohala Coast, HI, USA, January 2016.
- [43] J. Overeem, C. Robinson, H. Leijnse, G.-J. Steeneveld, B. K. Horn, and R. Uijlenhoet, "Crowdsourcing urban air temperatures from smartphone battery temperatures," *Geophysical Research Letters*, vol. 40, no. 15, pp. 4081–4085, 2013.
- [44] X. Su, H. Tong, and P. Ji, "Activity recognition with smartphone sensors," *Tsinghua Science and Technology*, vol. 19, no. 3, pp. 235–249, 2014.
- [45] C. Shepard, A. Rahmati, C. Tossell, L. Zhong, and P. Kortum, "LiveLab: measuring wireless networks and smartphone users in the field," *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 3, pp. 15–20, 2011.
- [46] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *ESANN 2013 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, Bruges, Belgium, April 2013.
- [47] J.-L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita, "Transition-aware human activity recognition using smartphones," *Neurocomputing*, vol. 171, pp. 754–767, 2016.
- [48] P. Barsocchi, A. Crivello, D. La Rosa, and F. Palumbo, "A multisource and multivariate dataset for indoor localization methods based on WLAN and geo-magnetic field fingerprinting," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, Calgary, Canada, October 2016.
- [49] R. Rawassizadeh, M. Tomitsch, K. Wac, and A. M. Tjoa, "UbiqLog: a generic mobile phone-based life-log framework," *Personal and Ubiquitous Computing*, vol. 17, no. 4, pp. 621–637, 2013.
- [50] R. Rawassizadeh, E. Momeni, C. Dobbins, P. Mirza-Babaei, and R. Rahnamoun, "Lesson learned from collecting quantified self information via mobile and wearable devices,"

- Journal of Sensor and Actuator Networks*, vol. 4, no. 4, pp. 315–335, 2015.
- [51] T. Dietterich, “Overfitting and undercomputing in machine learning,” *ACM computing surveys*, vol. 27, no. 3, pp. 326–327, 1995.
- [52] E. Lau, X. Liu, C. Xiao, and X. Yu, “Enhanced user authentication through keystroke biometrics,” *Computer and Network Security*, vol. 6, 2004.
- [53] Y. Meng, D. S. Wong, R. Schlegel, and L.-f. Kwok, “Touch gestures based biometric authentication scheme for touchscreen mobile phones,” in *Information Security and Cryptology. Inscrypt 2012*, M. Kutyłowski and M. Yung, Eds., vol. 7763 of Lecture Notes in Computer Science, pp. 331–350, Springer, Berlin, Heidelberg, 2012.
- [54] Y. Zhao, “Learning user keystroke patterns for authentication,” in *Proceeding of World Academy of Science, Engineering and Technology*, vol. 14, pp. 65–70, 2006.
- [55] G. Kambourakis, D. Damopoulos, D. Papamartzivanos, and E. Pavlidakis, “Introducing touchstroke: keystroke-based authentication system for smartphones,” *Security and Communication Networks*, vol. 9, no. 6, 554 pages, 2016.
- [56] N. Clarke, S. Furnell, B. Lines, and P. Reynolds, “Subscriber authentication for mobile phones using keystroke dynamics,” in *Proceedings of the Third International Network Conference (INC 2002)*, pp. 347–355, Plymouth, UK, 2002.
- [57] N. L. Clarke and S. M. Furnell, “Authenticating mobile phone users using keystroke analysis,” *International Journal of Information Security*, vol. 6, no. 1, pp. 1–14, 2007.
- [58] L. Cai and H. Chen, “TouchLogger: inferring keystrokes on touch screen from smartphone motion,” *Hot Topics in Security (HotSec)*, vol. 11, pp. 9–9, 2011.
- [59] T. Feng, Z. Liu, K.-A. Kwon et al., “Continuous mobile authentication using touchscreen gestures,” in *2012 IEEE Conference on Technologies for Homeland Security (HST)*, pp. 451–456, Waltham, MA, USA, November 2012.
- [60] N. Sae-Bae, K. Ahmed, K. Isbister, and N. Memon, “Biometric-rich gestures: a novel approach to authentication on multi-touch devices,” in *CHI '12 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 977–986, Austin, TX, USA, May 2012.
- [61] P. S. Teh, A. B. J. Teoh, and S. Yue, “A survey of keystroke dynamics biometrics,” *The Scientific World Journal*, vol. 2013, Article ID 408280, 24 pages, 2013.
- [62] X. K. Peacock and M. Wilkerson, “Typing patterns: a key to user identification,” *IEEE Security & Privacy Magazine*, vol. 2, no. 5, pp. 40–47, 2004.
- [63] H. Crawford, “Keystroke dynamics: characteristics and opportunities,” in *2010 Eighth International Conference on Privacy, Security and Trust*, pp. 205–212, Paris, France, August 2010.
- [64] W. Xu, Y. Shen, Y. Zhang, N. Bergmann, and W. Hu, “Gaitwatch: a context-aware authentication system for smart watch based on gait recognition,” in *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation - IoTDI '17*, pp. 59–70, Pittsburgh, PA, USA, 2017.
- [65] N. Al-Naffakh, N. Clarke, and F. Li, “Continuous user authentication using smartwatch motion sensor data,” in *Trust Management XII. IFIPTM 2018*, IFIP Advances in Information and Communication Technology, N. Gal-Oz and P. Lewis, Eds., pp. 15–28, 2018.
- [66] Z. Wang, C. Shen, and Y. Chen, “Handwaving authentication: unlocking your smartwatch through handwaving biometrics,” in *Biometric Recognition. CCBP 2017*, J. Zhou, Ed., vol. 10568 of Lecture Notes in Computer Science, pp. 545–553, Springer, Cham, 2017.
- [67] R. Mayrhofer and H. Gellersen, “Shake well before use: authentication based on accelerometer data,” in *Pervasive Computing. Pervasive 2007*, A. LaMarca, M. Langheinrich, and K. N. Truong, Eds., vol. 4480 of Lecture Notes in Computer Science, pp. 144–161, Springer, Berlin, Heidelberg, 2007.
- [68] H. Gascon, S. Uellenbeck, C. Wolf, and K. Rieck, “Continuous authentication on mobile devices by analysis of typing motion behavior,” in *Sicherheit 2014 – Sicherheit, Schutz und Zuverlässigkeit*, S. Katzenbeisser, V. Lotz, and E. Weippl, Eds., pp. 1–12, Gesellschaft für Informatik e.V, Bonn, 2014.
- [69] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song, “Touchalytics: on the applicability of touchscreen input as a behavioral biometric for continuous authentication,” *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 136–148, 2013.
- [70] S. Krishnamoorthy, L. Rueda, S. Saad, and H. Elmiligi, “Identification of user behavioral biometrics for authentication using keystroke dynamics and machine learning,” in *ICBEA '18 Proceedings of the 2018 2nd International Conference on Biometric Engineering and Applications*, pp. 50–57, Amsterdam, Netherlands, May 2018.
- [71] W.-H. Lee and R. Lee, “Implicit sensor-based authentication of smartphone users with smartwatch,” in *HASP 2016 Proceedings of the Hardware and Architectural Support for Security and Privacy 2016*, pp. 9:1–9:8, Seoul, Republic of Korea, June 2016.
- [72] C. Shen, Y. Li, Y. Chen, X. Guan, and R. A. Maxion, “Performance analysis of multi-motion sensor behavior for active smartphone authentication,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 48–62, 2018.
- [73] W. Lane and C. Manner, “The impact of personality traits on smartphone ownership and use,” *International Journal of Business and Social Science*, vol. 2, no. 17, 2011.
- [74] A. Ehrenberg, S. Juckes, K. M. White, and S. P. Walsh, “Personality and self-esteem as predictors of young people’s technology use,” *Cyberpsychology & Behavior*, vol. 11, no. 6, pp. 739–741, 2008.
- [75] J. G. Phillips, S. Butt, and A. Blaszczynski, “Personality and self-reported use of mobile phones for games,” *Cyberpsychology & Behavior*, vol. 9, no. 6, pp. 753–758, 2006.
- [76] S. Harada, D. Sato, H. Takagi, and C. Asakawa, “Characteristics of elderly user behavior on mobile multi-touch devices,” in *Human-Computer Interaction – INTERACT 2013. INTERACT 2013*, P. Kotzé, G. Marsden, G. Lindgaard, J. Wesson, and M. Winckler, Eds., vol. 8120 of Lecture Notes in Computer Science, pp. 323–341, Springer, Berlin, Heidelberg, 2013.
- [77] Y. Uzun, K. Bicakci, and Y. Uzunay, “Could we distinguish child users from adults using keystroke dynamics?,” 2015, <http://arxiv.org/abs/1511.05672>.
- [78] J. Goncalves, Z. Sarsenbayeva, N. van Berkel et al., “Tapping task performance on smartphones in cold temperature,” *Interacting with Computers*, vol. 29, no. 3, pp. 355–367, 2016.
- [79] Z. Sarsenbayeva, J. Goncalves, J. Garcia et al., “Situational impairments to mobile interaction in cold environments,” in *UbiComp '16 Proceedings of the 2016 ACM International*

- Joint Conference on Pervasive and Ubiquitous Computing*, pp. 85–96, Heidelberg, Germany, September 2016.
- [80] R. Giot and C. Rosenberger, “A new soft biometric approach for keystroke dynamics based on gender recognition,” *International Journal of Information Technology and Management*, vol. 11, no. 1-2, pp. 35–49, 2012.
- [81] B. Akis, M. Sorgente, and A. Starosta, *Typeguess: Using Mobile Typing Dynamics to Predict Age, Gender and Number of Fingers Used for Typing*, Standord University, 2014.
- [82] S. M. Kolly, R. Wattenhofer, and S. Welten, “A personal touch: recognizing users based on touch screen behavior,” in *PhoneSense '12 Proceedings of the Third International Workshop on Sensing Applications on Mobile Phones*, p. 1, Toronto, ON, Canada, November 2012.
- [83] T. Yan, D. Chu, D. Ganesan, A. Kansal, and J. Liu, “Fast app launching for mobile devices using predictive user context,” in *MobiSys '12 Proceedings of the 10th international conference on Mobile systems, applications, and services*, pp. 113–126, Low Wood Bay, Lake District, UK, June 2012.
- [84] R. LiKamWa, Y. Liu, N. D. Lane, and L. Zhong, “Moodscope: building a mood sensor from smartphone usage patterns,” in *MobiSys '13 Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pp. 389–402, Taipei, Taiwan, June 2013.
- [85] P. E. Dantcheva and A. Ross, “What else does your biometric data reveal? A survey on soft biometrics,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 3, pp. 441–467, 2016.
- [86] Y. Sun, M. Zhang, Z. Sun, and T. Tan, “Demographic analysis from biometric data: achievements, challenges, and new frontiers,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 2, pp. 332–351, 2018.
- [87] C. Bevan and D. S. Fraser, “Different strokes for different folks? Revealing the physical characteristics of smartphone users from their swipe gestures,” *International Journal of Human-Computer Studies*, vol. 88, pp. 51–61, 2016.
- [88] K. Shanavaz and P. Mythili, “A fingerprint-based hybrid gender classification system using genetic algorithm,” *International Journal of Computational Vision and Robotics*, vol. 6, no. 4, pp. 399–413, 2016.
- [89] S. Abdullah, A. Rahman, Z. Abas, and W. Saad, “Multilayer perceptron neural network in classifying gender using fingerprint global level features,” *Indian Journal of Science and Technology*, vol. 9, no. 9, 2016.
- [90] R. Soames and A. Evans, “Female gait patterns: the influence of footwear,” *Ergonomics*, vol. 30, no. 6, pp. 893–900, 1987.
- [91] H. Merrifield, “Female gait patterns in shoes with different heel heights,” *Ergonomics*, vol. 14, no. 3, pp. 411–417, 1971.
- [92] D. A. Johnson and M. M. Trivedi, “Driving style recognition using a smartphone as a sensor platform,” in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1609–1615, Washington, DC, USA, October 2011.
- [93] J. Dai, J. Teng, X. Bai, Z. Shen, and D. Xuan, “Mobile phone based drunk driving detection,” *Proceedings of the 4th International ICST Conference on Pervasive Computing Technologies for Healthcare*, 2010, pp. 1–8, Munchen, Germany, June 2010.
- [94] B. D. Martin, V. Addona, J. Wolfson, G. Adomavicius, and Y. Fan, “Methods for real-time prediction of the mode of travel using smartphone-based GPS and accelerometer data,” *Sensors*, vol. 17, no. 9, article 2058, 2017.
- [95] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, “Human activity recognition on smartphones using a multi-class hardware-friendly support vector machine,” in *Ambient Assisted Living and Home Care. IWAAL 2012*, J. Bravo, R. Hervás, and M. Rodríguez, Eds., vol. 7657 of Lecture Notes in Computer Science, pp. 216–223, Springer, Berlin, Heidelberg, 2012.
- [96] C. A. Ronao and S.-B. Cho, “Human activity recognition with smartphone sensors using deep learning neural networks,” *Expert Systems with Applications*, vol. 59, pp. 235–244, 2016.
- [97] Y. Chen and C. Shen, “Performance analysis of smartphone-sensor behavior for human activity recognition,” *Ieee Access*, vol. 5, pp. 3095–3110, 2017.
- [98] U. A. Abdulla, K. Taylor, M. Barlow, and K. Z. Naqshbandi, “Measuring walking and running cadence using magnetometers,” in *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, pp. 1458–1462, Liverpool, UK, July 2013.
- [99] K. Kunze, G. Bahle, P. Lukowicz, and K. Partridge, “Can magnetic field sensors replace gyroscopes in wearable sensing applications?,” in *International Symposium on Wearable Computers (ISWC) 2010*, pp. 1–4, London, UK, October 2010.
- [100] G. M. Weiss, J. W. Lockhart, T. T. Pulickal, P. T. McHugh, I. H. Ronan, and J. L. Timko, “Actitracker: a smartphone-based activity recognition system for improving health and well-being,” in *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 682–688, October 2016.
- [101] R. Gouveia, E. Karapanos, and M. Hassenzahl, “How do we engage with activity trackers?: a longitudinal study of Habito,” in *UbiComp '15 Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 1305–1316, Osaka, Japan, September 2015.
- [102] K. Lorincz, B.-r. Chen, G. W. Challen et al., “Mercury: a wearable sensor network platform for high-fidelity motion analysis,” in *SenSys '09 Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, pp. 183–196, Berkeley, CA, USA, November 2009.
- [103] Y. Lee, S. Iyengar, C. Min et al., “Mobicon: a mobile context-monitoring platform,” *Communications of the ACM*, vol. 55, no. 3, pp. 54–65, 2012.
- [104] G. Vavoulas, M. Pedititis, C. Chatzaki, E. G. Spanakis, and M. Tsiknakis, “The MobiFall dataset: fall detection and classification with a smartphone,” in *Communications of the ACM*, pp. 1218–1231, ACM, 2017.
- [105] A. O. Kansiz, M. A. Guvensan, and H. I. Turkmen, “Selection of time-domain features for fall detection based on supervised learning,” in *Proceedings of the World Congress on Engineering and Computer Science 2013*, vol. 2325, pp. 95–105, San Francisco, CA, USA, October 2013.
- [106] P. Fahmi, V. Viet, and C. Deok-Jai, “Semi-supervised fall detection algorithm using fall indicators in smartphone,” in *ICUIMC '12 Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication*, pp. 122:1–122:9, Kuala Lumpur, Malaysia, February 2012.
- [107] Z. Zhao, Y. Chen, S. Wang, and Z. Chen, “FallAlarm: smart phone based fall detecting and positioning system,” *Procedia Computer Science*, vol. 10, pp. 617–624, 2012.

- [108] G. M. Weiss, J. L. Timko, C. M. Gallagher, K. Yoneda, and A. J. Schreiber, "Smartwatch-based activity recognition: a machine learning approach," in *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pp. 426–429, Las Vegas, NV, USA, February 2016.
- [109] F. B. A. Ramos, A. Lorayne, A. A. M. Costa, R. R. de Sousa, H. O. Almeida, and A. Perkusich, "Combining smartphone and smartwatch sensor data in activity recognition approaches: an experimental evaluation," in *Proceedings of the 28th International Conference on Software Engineering and Knowledge Engineering*, pp. 267–272, San Francisco Bay, CA, USA, July 2016.
- [110] M. Susi, V. Renaudin, and G. Lachapelle, "Motion mode recognition and step detection algorithms for mobile phone users," *Sensors*, vol. 13, no. 2, pp. 1539–1562, 2013.
- [111] J. Chen, Y. Zhang, and W. Xue, "Unsupervised indoor localization based on smartphone sensors, iBeacon and Wi-Fi," *Sensors*, vol. 18, no. 5, 2018.
- [112] W. Kang and Y. Han, "SmartPDR: smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors Journal*, vol. 15, no. 5, pp. 2906–2916, 2015.
- [113] Z. Liu, L. Zhang, Q. Liu, Y. Yin, L. Cheng, and R. Zimmermann, "Fusion of magnetic and visual sensors for indoor localization: infrastructure-free and more effective," *IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 874–888, 2017.
- [114] S. Saha, S. Chatterjee, A. K. Gupta, I. Bhattacharya, and T. Mondal, "TrackMe-a low power location tracking system using smart phone sensors," in *2015 International Conference on Computing and Network Communications (CoCoNet)*, pp. 457–464, India, December 2015.
- [115] Y. Xuan, R. Sengupta, and Y. Fallah, "Making indoor maps with portable accelerometer and magnetometer," in *2010 Ubiquitous Positioning Indoor Navigation and Location Based Service*, pp. 1–7, Kirkkonummi, Finland, October 2010.
- [116] K. Muralidharan, A. J. Khan, A. Misra, R. K. Balan, and S. Agarwal, "Barometric phone sensors: more hype than hope!," in *HotMobile '14 Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*, pp. 12:1–12:6, Santa Barbara, CA, USA, February 2014.
- [117] S. Hyuga, M. Ito, M. Iwai, and K. Sezaki, "Estimate a user's location using smartphone's barometer on a subway," in *MELT '15 Proceedings of the 5th International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, pp. 2:1–2:4, Seattle, WA, USA, November 2015.
- [118] P. Zhou, M. Li, and G. Shen, "Use it free: instantly knowing your phone attitude," in *MobiCom '14 Proceedings of the 20th annual international conference on Mobile computing and networking*, pp. 605–616, Maui, HI, USA, September 2014.
- [119] M.-Z. Poh and Y. C. Poh, "Validation of a standalone smartphone application for measuring heart rate using imaging photoplethysmography," *Telemedicine and e-Health*, vol. 23, no. 8, pp. 678–683, 2017.
- [120] B. P. Yan, C. K. Chan, C. K. Li et al., "Resting and postexercise heart rate detection from fingertip and facial photoplethysmography using a smartphone camera: a validation study," *JMIR mHealth and uHealth*, vol. 5, no. 3, p. e33, 2017.
- [121] K. Qian, C. Wu, F. Xiao et al., "Acousticcardiogram: monitoring heartbeats using acoustic signals on smart devices," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, April 2018.
- [122] K.-H. Chen, W.-C. Tseng, K.-C. Liu, and C.-T. Chan, "Using gyroscopes and accelerometers as a practical rehabilitation monitor system after total knee arthroplasty," in *2015 IEEE MTT-S 2015 International Microwave Workshop Series on RF and Wireless Technologies for Biomedical and Healthcare Applications (IMWS-BIO)*, pp. 58–59, Taiwan, September 2015.
- [123] J. Guo, T. Smith, D. Messing, Z. Tang, S. Lawson, and J. H. Feng, "ARMStrokes: a mobile app for everyday stroke rehabilitation," in *ASSETS '15 Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility*, pp. 429–430, Lisbon, Portugal, October 2015.
- [124] E. J. Wang, W. Li, D. Hawkins, T. Gernsheimer, C. Norby-Slycord, and S. N. Patel, "HemaApp: noninvasive blood screening of hemoglobin using smartphone cameras," in *UbiComp '16 Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 593–604, Heidelberg, Germany, September 2016.
- [125] G. Zouridakis, T. Wadhawan, N. Situ et al., "Melanoma and other skin lesion detection using smart handheld devices," in *Mobile Health Technologies*, A. Rasooly and K. Herold, Eds., vol. 1256 of *Methods in Molecular Biology*, pp. 459–496, Humana Press, New York, NY, USA, 2015.
- [126] D. Ben-Zeev, E. A. Scherer, R. Wang, H. Xie, and A. T. Campbell, "Next-generation psychiatric assessment: using smartphone sensors to monitor behavior and mental health," *Psychiatric rehabilitation journal*, vol. 38, no. 3, pp. 218–226, 2015.
- [127] M. Drahansky, M. Dolezel, J. Urbanek, E. Brezinova, and T.-h. Kim, "Influence of skin diseases on fingerprint recognition," *Journal of Biomedicine and Biotechnology*, vol. 2012, Article ID 626148, 14 pages, 2012.
- [128] M. Drahansky, E. Brezinova, D. Hejtmankova, and F. Orsag, "Fingerprint recognition influenced by skin diseases," *International Journal of Bio-Science and Bio-Technology*, vol. 2, no. 4, 2010.
- [129] H. S. Kahn, M. Graff, A. D. Stein, and L. Lumey, "A fingerprint marker from early gestation associated with diabetes in middle age: the Dutch hunger winter families study," *International Journal of Epidemiology*, vol. 38, no. 1, pp. 101–109, 2008.
- [130] S. Srivastava and S. Rajasekar, "Comparison of digital and palmar dermatoglyphic patterns in diabetic and non-diabetic individuals," *IOSR Journal of Dental and Medical Sciences*, vol. 13, no. 7-2I, pp. 93–95, 2014.
- [131] C. Stevenson, C. West, and P. Pharoah, "Dermatoglyphic patterns, very low birth weight, and blood pressure in adolescence," *Archives of Disease in Childhood-Fetal and Neonatal Edition*, vol. 84, no. 1, pp. 18F–122, 2001.
- [132] G. M. Bhat, M. A. Mukhdoomi, B. A. Shah, and M. S. Ittoo, "Dermatoglyphics: in health and disease-a review," *International Journal of Research in Medical Sciences*, vol. 2, no. 1, pp. 31–37, 2017.
- [133] J. Fan, F. Han, and H. Liu, "Challenges of big data analysis," *National science review*, vol. 1, no. 2, pp. 293–314, 2014.
- [134] M. Brodie, E. Pliner, A. Ho et al., "Big data vs accurate data in health research: large-scale physical activity monitoring, smartphones, wearable devices and risk of unconscious bias," *Medical Hypotheses*, vol. 119, pp. 32–36, 2018.

- [135] L. Dennison, L. Morrison, G. Conway, and L. Yardley, "Opportunities and challenges for smartphone applications in supporting health behavior change: qualitative study," *Journal of medical Internet research*, vol. 15, no. 4, p. e86, 2013.
- [136] M. B. Del Rosario, K. Wang, J. Wang et al., "A comparison of activity classification in younger and older cohorts using a smartphone," *Physiological measurement*, vol. 35, no. 11, article 2269, 2286 pages, 2014.
- [137] N. Al-Naffakh, N. Clarke, F. Li, and P. Haskell-Dowland, "Unobtrusive gait recognition using smartwatches," in *2017 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pp. 1–5, September 2017.
- [138] T. Althoff, R. Sosič, J. L. Hicks, A. C. King, S. L. Delp, and J. Leskovec, "Large-scale physical activity data reveal worldwide activity inequality," *Nature*, vol. 547, no. 7663, pp. 336–339, 2017.
- [139] F.-X. Standaert, "Introduction to side-channel attacks," in *Secure Integrated Circuits and Systems, Integrated Circuits and Systems*, I. Verbauwhede, Ed., pp. 27–42, Springer, Boston, MA, USA, 2010.
- [140] M. Vuagnoux and S. Pasini, "Compromising electromagnetic emanations of wired and wireless keyboards," in *USENIX security symposium*, pp. 1–16, 2009.
- [141] Y. Berger, A. Wool, and A. Yeredor, "Dictionary attacks using keyboard acoustic emanations," in *CCS '06 Proceedings of the 13th ACM conference on Computer and communications security*, pp. 245–254, Alexandria, VA, USA, October 2006.
- [142] D. Foo Kune and Y. Kim, "Timing attacks on pin input devices," in *CCS '10 Proceedings of the 17th ACM conference on Computer and communications security*, pp. 678–680, Chicago, IL, USA, October 2010.
- [143] R. Tibbett, T. Volodine, S. Block, and A. Popescu, *DeviceOrientation Event Specification*, s Draft, w3c. github. io, apartado A, 1 edition, 2014.
- [144] L. Cai, S. Machiraju, and H. Chen, "Defending against sensor-sniffing attacks on mobile phones," in *MobiHeld '09 Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*, pp. 31–36, Barcelona, Spain, August 2009.
- [145] Y. Michalevsky, A. Schulman, G. A. Veerapandian, D. Boneh, and G. Nakibly, "Powerspy: location tracking using mobile device power analysis," in *24th USENIX Security Symposium*, pp. 785–800, Washington, DC, USA, August 2015.
- [146] J. Krumm and E. Horvitz, "LOCADIO: inferring motion and location from Wi-Fi signal strengths," in *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004*, pp. 4–13, 2004.
- [147] R. W. Ouyang, A. K.-S. Wong, C.-T. Lea, and V. Y. Zhang, "Received signal strength-based wireless localization via semidefinite programming," in *GLOBECOM 2009 - 2009 IEEE Global Telecommunications Conference*, pp. 1–6, November 2009.
- [148] T. Sohn, A. Varshavsky, A. LaMarca et al., "Mobility detection using everyday GSM traces," in *UbiComp 2006: Ubiquitous Computing. UbiComp 2006*, P. Dourish and A. Friday, Eds., vol. 4206 of Lecture Notes in Computer Science, pp. 212–224, Springer, Berlin, Heidelberg, 2006.
- [149] L. Yan, Y. Guo, X. Chen, and H. Mei, "A study on power side channels on mobile devices," in *Internetwork '15 Proceedings of the 7th Asia-Pacific Symposium on Internetwork*, pp. 30–38, Wuhan, China, November 2015.
- [150] M. Masoud, I. Jannoud, A. Ahmad, and H. Al-Shobaky, "The power consumption cost of data encryption in smartphones," in *2015 International Conference on Open Source Software Computing (OSSCOM)*, pp. 1–6, September 2015.
- [151] A. Mosenia, X. Dai, P. Mittal, and N. Jha, "PinMe: tracking a smartphone user around the world," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 4, no. 3, pp. 420–435, 2017.
- [152] S. Narain, T. D. Vo-Huu, K. Block, and G. Noubir, "Inferring user routes and locations using zero-permission mobile sensors," in *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 397–413, May 2016.
- [153] W. Diao, X. Liu, Z. Li, and K. Zhang, "No pardon for the interruption: new inference attacks on android through interrupt timing analysis," in *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 414–432, May 2016.
- [154] X. Zhou, S. Demetriou, D. He et al., "Identity, location, disease and more: inferring your secrets from android public resources," in *CCS '13 Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1017–1028, Berlin, Germany, November 2013.
- [155] R. Spreitzer, "Pin skimming: exploiting the ambient-light sensor in mobile devices," in *SPSM '14 Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, pp. 51–62, Scottsdale, AZ, USA, November 2014.
- [156] C. Xu, S. Li, G. Liu et al., "Crowd++: unsupervised speaker count with smartphones," in *UbiComp '13 Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pp. 43–52, Zurich, Switzerland, September 2013.
- [157] Y. Jeon, M. Kim, H. Kim, H. Kim, J. H. Huh, and J. W. Yoon, "I'm listening to your location! Inferring user location with acoustic side channels," in *WWW '18 Proceedings of the 2018 World Wide Web Conference*, pp. 339–348, Lyon, France, April 2018.
- [158] R. Schlegel, K. Zhang, X.-y. Zhou, M. Intwala, A. Kapadia, and X. Wang, "Soundcomber: a stealthy and context-aware sound Trojan for smartphones," in *Distributed System Security Symposium (NDSS)*, vol. 11, pp. 17–33, USA, 2011.
- [159] V. Srinivasan, S. Moghaddam, A. Mukherji, K. K. Rachuri, C. Xu, and E. M. Tapia, "Mobileminer: mining your frequent patterns on your phone," in *UbiComp '14 Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 389–400, Seattle, WA, USA, September 2014.
- [160] Y. Xu, M. Lin, H. Lu et al., "Preference, context and communities: a multi-faceted approach to predicting smartphone app usage patterns," in *ISWC '13 Proceedings of the 2013 International Symposium on Wearable Computers*, pp. 69–76, September 2013.
- [161] M. W. Janis, R. S. Kay, and A. W. Bradley, *European Human Rights Law: Text and Materials*, Oxford University Press, Oxford, UK, 2008.
- [162] E. Chin, A. P. Felt, V. Sekar, and D. Wagner, "Measuring user confidence in smartphone security and privacy," in *SOUPS '12 Proceedings of the Eighth Symposium on Usable Privacy and Security*, p. 1, Washington, DC, USA, July 2012.

- [163] C. Spensky, J. Stewart, A. Yerukhimovich et al., "Sok: privacy on mobile devices– its complicated," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 3, pp. 96–116, 2016.
- [164] A. Maiti, R. Heard, M. Sabra, and M. Jadliwala, "A framework for inferring combination lock codes using smartwatches," 2017, <http://arxiv.org/abs/1710.00217>.
- [165] A. Hojjati, A. Adhikari, K. Struckmann et al., "Leave your phone at the door: side channels that reveal factory floor secrets," in *CCS '16 Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 883–894, Vienna, Austria, October 2016.
- [166] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, "Touch me once and i know it's you!: implicit authentication based on touch screen patterns," in *CHI '12 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 987–996, Austin, TX, USA, May 2012.
- [167] T. R. Bhongale, S. S. Dhamnekar, A. K. Sanadi, S. K. Nandgave, A. A. Pawar, and S. A. Pardeshi, "Survey on tools and technologies applicable for mobile application development," *International Research Journal of Engineering and Technology*, vol. 5, no. 3, pp. 2985–2988, 2018.
- [168] B. C. Zapata, *Android Studio Application Development*, Packt Publishing Ltd, 2013.
- [169] S. M. Dol, R. Pise, P. Reure, R. Madde, and A. Valsangkar, "An interactive educational mobile application for system programming," *International Journal of Computer Applications*, vol. 108, no. 4, pp. 11–14, 2014.
- [170] A. Caliskan-Islam, R. Harang, A. Liu et al., "De-anonymizing programmers via code stylometry," in *24th USENIX Security Symposium*, Washington, DC, USA, August 2015.
- [171] A. Mylonas, V. Meletiadis, L. Mitrou, and D. Gritzalis, "Smartphone sensor data as digital evidence," *Computers & Security*, vol. 38, pp. 51–75, 2013.
- [172] M. Masoud, Y. Jaradat, I. Jannoud, and O. Heyasat, "A measurement study of the quality of zero line programming technique for smartphones applications," *International Journal on Communications Antenna and Propagation (IRECAP)*, vol. 5, no. 4, 2015.
- [173] S. Papadakis, M. Kalogiannakis, V. Orfanakis, and N. Zaranis, "Novice programming environments. Scratch & App Inventor: a first comparison," in *IDEE '14 Proceedings of the 2014 Workshop on Interaction Design in Educational Environments*, p. 1, Albacete, Spain, June 2014.
- [174] M. AppInventor, "Mit App Inventor," 2015, Technical report, <http://appinventor.mit.edu/explore>.
- [175] B. Xie, I. Shabir, and H. Abelson, "Measuring the usability and capability of app inventor to create mobile applications," in *PROMOTO 2015 Proceedings of the 3rd International Workshop on Programming for Mobile and Touch*, pp. 1–8, Pittsburgh, PA, USA, October 2015.
- [176] M. N. Jivani, "GSM based home automation system using app-inventor for android mobile phone," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, vol. 3, no. 9, pp. 12121–12128, 2014.
- [177] C. Keung, A. Lee, S. Lu, and M. O'Keefe, "BunnyBolt: a mobile fitness app for youth," in *IDC '13 Proceedings of the 12th International Conference on Interaction Design and Children*, pp. 585–588, New York, NY, USA, June 2013.
- [178] F. M. Kundi, A. Habib, A. Habib, and M. Z. Asghar, "Android-based health care management system," *International Journal of Computer Science and Information Security*, vol. 14, no. 7, p. 77, 2016.
- [179] F. Salamone, L. Belussi, L. Danza, M. Ghellere, and I. Meroni, "An open source "smart lamp" for the optimization of plant systems and thermal comfort of offices," *Sensors*, vol. 16, no. 3, p. 338, 2016.
- [180] I. Amerini, P. Bestagini, L. Bondi, R. Caldelli, M. Casini, and S. Tubaro, "Robust smartphone fingerprint by mixing device sensors features for mobile strong authentication," *Electronic Imaging*, vol. 2016, no. 8, pp. 1–8, 2016.
- [181] A. Das, N. Borisov, and M. Caesar, "Exploring ways to mitigate sensorbased smartphone fingerprinting," 2015, <http://arxiv.org/abs/1503.01874>.
- [182] N. Zheng, K. Bai, H. Huang, and H. Wang, "You are how you touch: user verification on smartphones via tapping behaviors," in *2014 IEEE 22nd International Conference on Network Protocols*, vol. 14, pp. 221–232, October 2014.
- [183] E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R. R. Choudhury, "TapPrints: your finger taps have fingerprints," in *MobiSys '12 Proceedings of the 10th international conference on Mobile systems, applications, and services*, pp. 323–336, Low Wood Bay, Lake District, UK, June 2012.
- [184] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "Accessory: password inference using accelerometers on smartphones," in *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, p. 9, San Diego, CA, USA, February 2012.
- [185] C. D. Barclay, J. E. Cutting, and L. T. Kozlowski, "Temporal and spatial factors in gait perception that influence gender recognition," *Perception & Psychophysics*, vol. 23, no. 2, pp. 145–152, 1978.
- [186] G. M. Weiss and J. W. Lockhart, "Identifying user traits by mining smart phone accelerometer data," in *SensorKDD '11 Proceedings of the Fifth International Workshop on Knowledge Discovery from Sensor Data*, pp. 61–69, San Diego, CA, USA, August 2011.
- [187] N. Bhargava, G. Sharma, R. Bhargava, and M. Mathuria, "Decision tree analysis on j48 algorithm for data mining," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 6, pp. 1114–1119, 2013.
- [188] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2016.
- [189] O. Miguel-Hurtado, S. V. Stevenage, C. Bevan, and R. Guest, "Predicting sex as a soft-biometrics from device interaction swipe gestures," *Pattern Recognition Letters*, vol. 79, pp. 44–51, 2016.
- [190] P. Gnanasivam and D. S. Muttan, "Fingerprint gender classification using wavelet transform and singular value decomposition," 2012, <http://arxiv.org/abs/1205.6745>.
- [191] H. Bojinov, Y. Michalevsky, G. Nakibly, and D. Boneh, "Mobile device identification via sensor fingerprinting," 2014, <http://arxiv.org/abs/1408.1416>.
- [192] J. R. Corripio, D. A. Gonzalez, A. S. Orozco, L. G. Villalba, J. Hernandez-Castro, and S. J. Gibson, "Source smartphone identification using sensor pattern noise and wavelet transform," in *5th International Conference on Imaging for Crime Detection and Prevention (ICDP 2013)*, 2013.

- [193] A. Kurtz, H. Gascon, T. Becker, K. Rieck, and F. Freiling, "Fingerprinting mobile devices using personalized configurations," *Proceedings on Privacy Enhancing Technologies*, vol. 2016, no. 1, pp. 4–19, 2016.
- [194] M.-J. Tsai, C.-L. Lai, and J. Liu, "Camera/mobile phone source identification for digital forensics," in *IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, vol. 2, 2007.
- [195] A. L. S. Orozco, J. R. Corripio, L. J. G. Villalba, and J. C. H. Castro, "Image source acquisition identification of mobile devices based on the use of features," *Multimedia Tools and Applications*, vol. 75, no. 12, pp. 7087–7111, 2016.
- [196] S. Dey, N. Roy, W. Xu, R. R. Choudhury, and S. Nelakuditi, "AccelPrint: imperfections of accelerometers make smartphones trackable," in *NDSS Symposium 2014*, February 2014.
- [197] Z. Xu, K. Bai, and S. Zhu, "TapLogger: inferring user inputs on smartphone touchscreens using on-board motion sensors," in *WISEC '12 Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*, pp. 113–124, Tucson, Arizona, USA, April 2012.
- [198] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, pp. 1–27, 2011.
- [199] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [200] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard acoustic emanations revisited," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, no. 1, pp. 1–26, 2009.
- [201] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: the Munich versatile and fast open-source audio feature extractor," in *MM '10 Proceedings of the 18th ACM international conference on Multimedia*, pp. 1459–1462, Firenze, Italy, October 2010.
- [202] M. I. Jordan, *An Introduction to Probabilistic Graphical Models*, Springer, 2003.
- [203] K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X. Wang, "Soundminer: a stealthy and context-aware sound Trojan for smartphones," in *Proceedings of the 18th Annual Network and Distributed System Security Symposium*, 2011.
- [204] L. J. Latecki, Q. Wang, S. Koknar-Tezel, and V. Megalooikonomou, "Optimal subsequence bijection," in *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pp. 565–570, October 2007.
- [205] M. Muller, *Information Retrieval for Music and Motion*, Springer, 2 edition, 2007.

