

A Device Fingerprint Extraction Method Based on Smart Phone Sensor

Technical Field

This invention belongs to the field of mobile terminal. The device fingerprint extraction technology based on smart phone sensor mainly relates to the collection, calculation, verification, calibration and other operations of sensor information of smart phone, in particular to the calculation and verification of bias gain of gyroscope, accelerometer and magnetometer as well as the generation and verification of smart phone device fingerprint.

Background Art

The sensor is an important part of smart phones. Each smart phone contains many sensor elements such as cameras, microphones, light sensors to GPS, gyroscopes and accelerometers which are all based on the technology of MEMS (Micro Electro Mechanical Systems). For a good sensor, accuracy is very important, so device manufacturers usually use factory calibration to compensate for the deviation in the manufacturing process. The calibration data of each sensor can be calculated by analyzing the output data of the sensor and used as the fingerprint of a device.

At present, the smart phones mainly include two operating systems: Android and iOS. The traditional Android system device fingerprints are mainly generated based on the information of IMEI, MAC and ANDROID_ID for extraction and usually be one or more elements; the traditional iOS system device fingerprints are mainly generated based on the values of IDFA, IDFV and OpenUDID for extraction. As Android system is upgrading and Google restricts the access to the values of IMEI and MAC, the traditional fingerprint access can no longer meet the requirements of unique identification devices, and the access to these information requires permission application, which will significantly affect the user experience. In addition, in the low

version of Android, these values are also subject to risks such as tampering and misuse; the current iOS system also cannot obtain the device unique fingerprints in the event the ex-factory setting is fully restored, restricting the attribution and traceability of devices.

Contents of the Invention

This invention is realized by the technical scheme as follows.

With the purpose to solve these problems, this invention proposes a device fingerprint extraction technology based on smart phone sensor information. To achieve the above purpose, the first embodiment of this invention proposes a device fingerprint extraction method based on smart phone sensor, which includes the steps as follows:

Step I: data acquisition;

Step II: data preprocessing;

Step III: ADC value recovery;

Step IV: gain matrix estimation;

Step V: validity check

Step VI: Device fingerprint ID generation

With the purpose to achieve the above purpose, the second embodiment of this invention proposes an electronic device, comprising the memory, the processor and the computer program stored on the memory and capable of running on the processor, which will be executed to realize the method mentioned when the processor operates the computer program.

With the purpose to achieve the above purpose, the third embodiment of this invention proposes a computer-readable storage medium on which a computer program is stored, which will be executed to realize the method mentioned when the processor operates

the computer program.

This invention has the advantages that: it can extract device fingerprint exactly, and can uniquely identify device when the device fingerprint ID doesn't change when certain devices are sampled for ex-factory setting recovery, machine upgrade and time point testing.

Illustrations

By reading the details of the selected execution modes below, the common technicians of this field will be clear of all advantages of this invention. The figures can only be used to show the purpose of the selected execution modes rather than to restrict this invention. Over the whole figures, the same reference symbols represent the same parts. In the figures:

Fig. 1 shows the flow chart of a device fingerprint extraction method based on smartphone sensor according to the execution modes of this invention;

Fig. 2 shows the typical diagram of sensor acquisition data according to the execution modes of this invention;

Fig. 3 shows the structural diagram of an electronic device provided by an embodiment of this invention;

Fig. 4 shows the schematic diagram of a computer medium provided by an embodiment of this invention.

Embodiments

We will describe the typical execution modes in detail with the reference to the figures. Though the figures show the typical execution modes of this invention, we shall understand that this invention can be realized in all forms rather than be restricted by the execution modes herein. On the contrary, these execution modes are provided with the purpose to make this invention more understandable and transmit the scope of this

invention to the technicians of this field.

Noted that unless otherwise specified, the technical terms or scientific terms used in this invention shall be the general meaning understood by the technicians of this field.

In addition, the terms "first", "second" and the like are used to distinguish different objects rather than to describe a particular order. In addition, the terms "include", "have" and their deformations are intended to cover the non-exclusive inclusions. For example, the processes, methods, systems, products or devices that contain a series of steps or units are not limited to the listed steps or units, but optionally also include the steps or units that are not listed, or optionally include other steps or units that are fixed to these processes, methods, products or devices.

During the work, this invention can infer the ex-factory calibration data through the outputs from gyroscope, accelerometer, magnetometer and other sensors and construct a valid device fingerprint which can effectively identify the device uniqueness with these calibration data. The reason why this invention uses the sensor information is that the access to these data needn't permission application and is available in both Android and iOS systems with high efficiency, the acquisition of sample data can be completed within 1s and the fingerprint information extracted will not change as time goes and along with the ex-factory setting recovery, update, upgrade and other operations. This invention proves that the method hereunder can generate the unique fingerprint of a device.

The method adopted in this invention is to infer the ex-factory calibration data of each device according to the outputs of gyroscope, accelerometer, magnetometer and other sensors. MEMS sensors usually convert analog quantity into digital quantity by ADC and store it in register. For three-axis motion sensor, the outputs of ADC are generally expressed as follows: $A = [A_x, A_y, A_z]^T$. According to the calibration model of standard sensors, the standard outputs of sensors can be expressed as follows:

$$\begin{pmatrix} O_x \\ O_y \\ O_z \end{pmatrix} = \begin{bmatrix} P_x & 0 & 0 \\ 0 & P_y & 0 \\ 0 & 0 & P_z \end{bmatrix} \begin{bmatrix} N_{xx} & N_{xy} & N_{xz} \\ N_{yx} & N_{yy} & N_{yz} \\ N_{zx} & N_{zy} & N_{zz} \end{bmatrix} \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} + \begin{bmatrix} D_x \\ D_y \\ D_z \end{bmatrix}$$

Wherein, $P_i \in P$ refers to the scale factor; $N_{ij} \in N$ refers to the non-orthogonality between axes i and j; $B_i \in B$ refers to the standard deviation; A refers to the sensitivity or gain of the sensor, which is defined as the ratio between the output signal and the measurement attribute, and the nominal gain of the sensor is the expected working sensitivity of the sensor. In this invention, M is used to represent the nominal gain of the sensor. If the sensor is ideal, its scale matrix P and non-orthogonality matrix N shall respectively be $M \cdot I$ and I, wherein I is the unit matrix. So the above equation can be further simplified as:

$$O = HA + D$$

Wherein, $H = PN$ represents nominal gain. The calibration standard of the sensor has multi latitudes, including high-precision device, multi position, Kalman filter and vision based. Manufacturers usually choose to only calibrate the offset vector to reduce costs. After the factory calibration is completed, these calibration data will be written into the hardware storage, and will not change as time goes nor readily rewritten.

After the device is calibrated in the factory, the gain matrix H and bias matrix D shall be unique, so any one of these two data can be recovered as the fingerprint of the device. The embodiments of this invention describe the recovery of the gain matrix H. With the purpose to recover the gain matrix, the invention needs to know the nominal gain of each sensor device which can be generally found in the sensor device specification of the manufacturer, for example, the nominal gain of the gyroscope of HUAWEI Mate 30 is 60 MDPs, that of of iPhone X is 61 MDPs and that of Google Pixel3XL is 64 mdps. The bias matrix D can be eliminated by the difference between the two sensors. The actual gain matrix and the ideal gain matrix are very similar, but not equal.

The basic method for generating sensor fingerprint of this invention includes six steps:

data acquisition, data preprocessing, ADC value recovery, gain matrix estimation, validity check, and device fingerprint ID generation. When the device is static or slowly moving, it can generate sensor fingerprint effectively. The invention will take the fingerprint generated by gyroscope data as an example below.

Embodiment 1

Step I: Data Acquisition

First of all, we can write an application app and acquire gyroscope sensor information on the device. Through the test of this invention, the acquisition efficiency on different models of Android device is different. About 400 groups of data can be acquired in HUAWEI Mate 30 one second and about 100 groups in GooglePixel3XL one second, and about 110 groups in iOS device one second. It is enough to acquire 100 groups of data herein. $O = [O_0, O_1, \dots, O_{n-1}]$ is used to identify the number of single groups of data and $O_i = [O_x, O_y, O_z]^T$ to represent the access to different axes.

Step II: Data Preprocessing

After obtaining the data, we can firstly carry out the differential calculation on the data to eliminate the offset matrix error D, and get ΔO by the equation as follows:

$$\Delta O = [O_1 - O_0, O_2 - O_1, \dots, O_{n-1} - O_{n-2}]$$

Step III: ADC value recovery

In this step, this invention mainly recovers ΔA which is the difference among successive ADCs. ΔA can be obtained from the $\Delta A = H^{-1} \Delta O$. H^{-1} is the inverse matrix of the actual gain matrix H, but the actual gain matrix is unknown herein, so the nominal gain can be temporarily replaced by the actual gain value in this invention:

$H_0 = M_H \cdot I$. M_H is the nominal gain of the gyroscope, and the value can be obtained

from the manufacturer. Herein, this invention only takes the integer value of ΔA , and it can be estimated by the equation as follows:

$$\tilde{\Delta A} = \text{round}(H_0^{-1} \Delta O) \quad (1)$$

At this time, H_0 is not equal to H but an estimated value, so $\tilde{\Delta A}$ is not really the data to be obtained. This invention also needs to eliminate the invalid value through the equation as follows:

$$E_{\Delta A}^r = \left| \tilde{\Delta A} - H_0^{-1} \Delta o \right| \quad (2)$$

With the purpose to assure the accuracy of the data, it is necessary to eliminate the data beyond the threshold value. At this time, the threshold value can be set to 0.1. The values can be eliminated according to the ADC values and the last remaining ADC value can be regarded as real data. At this time, the absolute value of ΔA is very small, and can be only used in the state of static or slow movement.

Step IV: Gain Matrix Estimation

After obtaining the ADC value ΔA , this invention can estimate the nominal gain $\tilde{H}$ by the equation as follows:

$$\tilde{H} = \Delta O / \Delta A$$

In this case, the matrix right division will return the least square solution of H, namely

$$\tilde{H} \Delta A = \Delta O.$$

Step V: Validity Check

With the purpose to quantify the deviation of $\tilde{H}$ from the true H, this invention calculates by the equation as follows:

$$E^e = std\left(\Delta O - \tilde{H} \Delta A\right)$$

The std herein is the standard deviation function. If the error is very small, the value of $\tilde{H}$ will be very close to the true H.

Herein, we take HUAWEI Mate 30 as an example, and will obtain the value of $\tilde{H}$ as follows:

$$\begin{bmatrix} 0.001065797050075 & -0.000010120011969 & -0.000004793689880 \\ 0.000002929477149 & 0.001071656004373 & 0.000003994741566 \\ 0.000007989483133 & 0.000006924218716 & 0.001070058107746 \end{bmatrix}$$

The ADC resolution of Mate30 is 16 bits. This invention can convert rds to dps and estimate the integer value:

$$\tilde{H} \cdot 2^{16} \cdot 180/\pi = \begin{bmatrix} 4002.000000021879 & -37.99999999747552 & -17.999999998606565 \\ 10.99999999956567 & 4024.000000021010 & 14.999999996335514 \\ 29.99999999642596 & 25.9999999965612 & 4018.0000000202235 \end{bmatrix}$$

We can see that in this case, these numbers are very close to integers, so we can round them to get the true gain matrix H. With the purpose to ensure reliability, this invention needs to calculate the error:

$$E_H^r = \left| \tilde{H}^d \cdot 2^{16} - round\left(\tilde{H}^d \cdot 2^{16}\right) \right|$$

The $\tilde{H}^d$ herein is the value of $\tilde{H}$ when its unit is dps. At this time, the error range of E obtained can be 0.01. When the error is greater than this value, it means that the group of data is unavailable, and the current device may be in a moving state. In this case, this invention needs to re-acquire data to calculate until the error reaches a certain degree.

Step VI: Device Fingerprint ID Generation

The generation of fingerprint ID can be obtained according to the following formula:

$$DeviceID = round\left(\tilde{H}^d \cdot 2^{16}\right) - round(H_0^d \cdot 2^{16})$$

It can be said that the DeviceID is the gain matrix H after the nominal gain is extracted in the unit of dps. The DeviceID of HUAWEI Mate 30 is exemplified as follows:

$$DeviceID = \begin{bmatrix} 4 & -38 & -18 \\ 11 & 27 & 15 \\ 30 & 26 & 22 \end{bmatrix}$$

The idea and process of generating device fingerprint ID of gyroscope are basically described herein. Generally speaking, these steps are realizable, but they require that the device is in a state of static or slow movement when it is measured. With the purpose to solve this problem, this invention has studied an improvement scheme which solves the device fingerprint extraction when the device is moving. See the embodiment 2 as follows.

Embodiment 2

In the Formula I in Step III, this invention estimates ΔA by $H_0 = M_H \cdot I$, but because H_0 is not equal to H, the rounded value of ΔA is not equal to the actual value of ΔA . Generally, as the element value in H_0 increases, the difference between ΔO and $H_0 \cdot \Delta A$ will increase, resulting in incorrect rounding. In the improvement method, in addition to data preprocessing, ADC value recovery and gain matrix estimation, this invention uses the same scheme as that of the basic steps. Herein, this invention does not directly use ΔO to input algorithm, but uses the data within different ranges to update and iterate H. The specific steps are as follows:

Step I: Data Acquisition

First of all, we can write an application app to acquire gyroscope sensor information on the device. After the test by this invention, the acquisition efficiency of different models of Android device is different. About 400 groups of data can be acquired in HUAWEI Mate 30 one second, about 100 groups in GooglePixel3XL one second, and about 110 groups in iOS device one second. It is enough to acquire 100 groups of data herein. This

invention is used to identify the number of single groups of data and to represent the access to different axes.

Step II: Data Preprocessing

Herein, this invention can firstly generate data with a smaller value from ΔO , because the smaller value is unlikely to exceed the rounded error, so this invention can sort the elements in ΔO , and then remove the difference between adjacent elements, so the result value will be very small. The data obtained can be expressed as $O = [O_0, O_1, \dots, O_{n-1}]$, wherein, $O_i = [O_x, O_y, O_z]^T$ represents single axis data. First, the values of ΔO_{ix} can be sorted in ascending order, and $[\Delta O]_x$ represents the sorted array. Similarly, this invention can sort the values of $[\Delta O]_y$ and $[\Delta O]_z$, and output the results, and then calculate as follows:

$$\Delta\Delta O = [\text{diff}([\Delta O]_x) \text{diff}([\Delta O]_y) \text{diff}([O]_z)] \quad (3)$$

The diff function uses different continuous vectors in the matrix. For example, the calculation formula of $\text{diff}(\Delta O)$ is:

$$\text{diff}(\Delta O) = [\Delta O_1 - \Delta O_0, \dots, \Delta O_{n-1} - \Delta O_{n-2}]$$

By subtracting similar vectors, $\Delta\Delta O$ contains more data of smaller values. According to the Equation 1, this invention can obtain:

$$H^{-1} \text{diff}(\Delta O) = \text{diff}(\Delta A)$$

Herein, the values of $\text{diff}(\Delta A)$ are all integers. Combined with the Formula 3, we can see that the values of HA shall only contain integers. Therefore, this invention can directly add $\Delta\Delta O$ to ΔO , and can perform this extension repeatedly to generate more smaller values. Then, according to the range of values, this invention can generate

multiple batches of data from the extended data and update them. The absolute values of all elements are lower than the products of programming gains, that is to say:

$$\Delta O^{\varepsilon i} = \{\Delta O_j \in \Delta O \mid \max(|\Delta O_j|) < (\varepsilon_i + 0.5)M_H\}$$

We can calculate as such batches and then double the value of each batch until it is the same as the value of ΔO . Then we can calculate $\Delta O^{\varepsilon i}$ step by step and update the value of H.

Step III: ADC Value Recovery

Herein, this invention does not use H_0 to estimate ΔA and round the error $E_{\Delta A}^r$, but uses H_0 of the initial value to replace $\tilde{H}$, so the above Formulas 1 and 2 shall be updated as follows:

$$\begin{aligned}\tilde{\Delta A}^{\varepsilon i} &= \text{round}\left(\tilde{H}^{-1} \Delta O^{\varepsilon i}\right) \\ E_{\Delta A^{\varepsilon i}}^r &= \left| \tilde{\Delta A}^{\varepsilon i} - \tilde{H}^{-1} \Delta O^{\varepsilon i} \right|\end{aligned}$$

Step IV: Gain Matrix Estimation

This step is mainly to update $\tilde{H}$ from $\Delta O^{\varepsilon i}$ zone of each batch of data. The main difference between the improvement method and the basic method is that the former can return to the data preprocessing step after the gain matrix estimation. After each update of $\tilde{H}$, the algorithm will check whether $\Delta O^{\varepsilon i} = \Delta O$ is tenable. If yes, it means that this invention has processed all the output data and passed the estimated $\tilde{H}$ to the validity check process. Otherwise, the algorithm will return to the pre-processing stage by using the updated $\tilde{H}$, and double filter ordinal for the next batch of data processing.

The improvement method is to iterate from smaller range of data to the whole data set, and estimate the output value of ADC through the estimated value, thus reducing the error of each estimation, so the device ID can be generated even when the device is moving. The basic method can show the feasibility of the sensor fingerprint, and the improvement method is applicable to actual scenarios and more reliable, which is the preferred way to obtain the device fingerprint in the actual situation.

Step V: Validity Check

With the purpose to quantify the deviation of $\tilde{H}$ from the true H, this invention calculates by the equation as follows:

$$E^e = std(\Delta O - \tilde{H} \Delta A)$$

The std herein is the standard deviation function. If the error is very small, the value of $\tilde{H}$ will be very close to the true H.

Herein, we take HUAWEI Mate 30 as an example, and will obtain the value of $\tilde{H}$ as follows:

$$\begin{bmatrix} 0.001065797050075 & -0.000010120011969 & -0.000004793689880 \\ 0.000002929477149 & 0.001071656004373 & 0.000003994741566 \\ 0.000007989483133 & 0.000006924218716 & 0.001070058107746 \end{bmatrix}$$

The ADC resolution of Mate30 is 16 bits. This invention can convert rds to dps and estimate the integer value:

$$\tilde{H} \cdot 2^{16} \cdot 180/\pi = \begin{bmatrix} 4002.000000021879 & -37.99999999747552 & -17.999999998606565 \\ 10.99999999956567 & 4024.000000021010 & 14.999999996335514 \\ 29.99999999642596 & 25.9999999965612 & 4018.0000000202235 \end{bmatrix}$$

We can see that in this case, these numbers are very close to integers, so we can round them to get the true gain matrix H. With the purpose to ensure reliability, this invention needs to calculate the error:

$$E_H^r = \left| \tilde{H}^d \cdot 2^{16} - \text{round}(\tilde{H}^d \cdot 2^{16}) \right|$$

The $\tilde{H}^d$ herein is the value of $\tilde{H}$ when its unit is dps. At this time, the error range of E obtained can be 0.01. When the error is greater than this value, it means that the group of data is unavailable, and the current device may be in a moving state. In this case, this invention needs to re-acquire data to calculate until the error reaches a certain degree.

Step VI: Device Fingerprint ID Generation

The generation of fingerprint ID can be obtained according to the following formula:

$$DeviceID = \text{round}(\tilde{H}^d \cdot 2^{16}) - \text{round}(H_0^d \cdot 2^{16})$$

It can be said that the DeviceID is the gain matrix H after the nominal gain is extracted in the unit of dps. The DeviceID of HUAWEI Mate 30 is exemplified as follows:

$$DeviceID = \begin{bmatrix} 4 & -38 & -18 \\ 11 & 27 & 15 \\ 30 & 26 & 22 \end{bmatrix}.$$

Test Results

After real device test, this invention found different device manufacturers take insufficient effort in the calibration of sensors. The manufacturers will not conduct calibration and verification of MEMS for cheap and low-end smart phones while certain Android manufacturers only conduct calibration on certain types such as HUAWEI Mate and Google Pixel. But the factories of iOS devices have conducted calibration since the time of iPhone4 and iPad3. This invention can accurately extract device fingerprints for the devices with the original sensors calibrated by factories. According to the statistics of this invention, this invention can extract the device fingerprint IDs with the accuracy rate of 100% after it tests 5,680 Android devices and 4,675 iOS devices. In addition, certain devices have been sampled for ex-factory setting recovery, machine upgrading and time point test, showing no change in the device fingerprint IDs

and uniqueness in device identification.

The execution modes of this invention also provide the electronic devices corresponding to the device fingerprint extraction method based on smart phone sensor provided by the aforesaid execution modes so as to execute the device fingerprint extraction method based on smart phone sensor. Such devices may be mobile phones, tablet computers and cameras, which are not stipulated in the embodiments of this invention.

With the reference to Fig. 3 which is the schematic diagram of the electronic devices provided by certain execution modes of this invention, the electronic device 2 comprises the processor 200, the memory 201, the bus 202 and the communication interface 203, and the processor 200, communication interface 203 and the memory 201 are connected through the bus 202; the memory 201 stores the computer program which can run in the processor 200 and the processor 200 will execute the device fingerprint extraction method based on smart phone sensor provided by any execution mode of this invention when it operates the computer program.

Thereof, the memory 201 may contain high-speed random access memory (RAM) and/or non-volatile memory which may be minimum one disk memory. The system network element may be communicated with minimum the other network element through minimum one communication interface 203 (wire or wireless), making the Internet, WAN, local network and MAN available.

The bus 202 may be ISA bus, PCI bus and EISA bus. The bus can be divided into address bus, data bus, control bus, etc. The memory 201 is used for storing programs, and the processor 200 will execute the programs after receiving the execution instructions. The device fingerprint extraction method based on smart phone sensor disclosed in any execution mode of this invention can be applied to or executed by the processor 200.

The processor 200 may be a kind of integrated circuit chip with signal processing

capability. During the execution, each step of the above method can be completed through the integrated logic circuit of the hardware or the instruction in the form of software in the processor 200. The above processor 200 can be general-purpose processor, comprising central processing unit (CPU), network processor (NP), etc.; or a digital signal processor (DSP), ASIC, FPGA or other programmable logic device, discrete gate or transistor logic device, and discrete hardware component, which can realize or execute all methods, steps and logic block diagrams in the embodiments of this invention. The general-purpose processor may be a microprocessor or any conventional processor, which can directly present the completion by the hardware decode processor or by the module of hardware and software in the decode processor combined with the steps of the methods disclosed in the embodiments of this invention. The software module can lie in RAM, FM, ROM, ROMP, EEPROM, MTRR and other mature storage mediums of this field which lie in the memory 201. The processor 200 will read the information of the memory 201 and complete the steps of the above methods combined with its hardware.

The electronic devices provided by the embodiments of this invention and the device fingerprint extraction method based on the smartphone sensor provided by embodiments of this invention are of the same inventive concept, and have the same beneficial effect as the method adopted, operated or realized.

The execution modes of this invention also provide a kind of computer-readable mediums corresponding to the device fingerprint extraction method based on the smart phone sensor provided by the aforesaid execution modes. With reference to the Fig. 4, the computer-readable storage medium is CD30 with the computer program (i.e. program product) and will execute the device fingerprint extraction method based on the smart phone sensor provided by any aforesaid execution modes when the computer program is executed by the processor.

Noted that the examples of the computer-readable storage mediums can also include without limitation to, PRAM, SRAM, DRAM, RAM, ROM, EEPROM, FM or other

optical and magnetic storage mediums, which is not described herein.

The computer-readable mediums provided by the embodiments of this invention and the device fingerprint extraction method based on the smartphone sensor provided by embodiments of this invention are of the same inventive concept, and have the same beneficial effect as the method adopted, operated or realized by the App stored.

In the description of the specification, the reference terms "an embodiment", "certain embodiments", "examples", "specific examples", or "certain examples" mean the minimum one embodiment or example contained in this invention combined with the specific features, structures, materials or characteristics described this embodiment or example. In this specification, the schematic expression of the above terms does not have to be directed to the same embodiment or example. Moreover, the specific features, structures, materials or characteristics described may be combined in an appropriate manner in any one or more embodiments or examples. In addition, without contradiction, the technicians of this field can combine and assemble different embodiments or examples described in this specification and features of different embodiments or examples.

In addition, the terms "first" and "second" are used to describe purposes only and cannot be understood as indicating or implying relative importance or implying the number of indicated technical features. Thus, the features defined as "first" or "second" may include minimum one such feature, either explicitly or implicitly. In the description of this invention, "multiple" means minimum two, such as two, three, etc., unless otherwise specifically defined.

Any process or method in the flowchart or described in other ways herein can be understood as representing a module, fragment or part of code including one or more executable instructions for implementing the steps of a custom logic function or process, and the scope of the selected embodiments of this invention includes additional implementation, which may follow the sequence of showing or discussion. The

functions can be executed in basic synchronous way or by inverse sequence, which shall be understood by the technicians of the field for the embodiments of this invention.

The logics and / or steps represented in a flowchart or otherwise described herein, for example, the priority list of the executable instructions considered for realizing the logic functions can be realized in any computer-readable medium to serve the instruction execution systems, units or devices (e.g. systems based on computer, systems with processor or other systems which can take instructions for instruction execution systems, units or devices and execute these instructions), or work in combination with these instruction execution systems, units or devices. In terms of of this specification, "computer-readable medium" may be any unit that may contain, store, communicate, propagate or transmit programs for use by or in combination with instruction execution systems, units or devices. A more specific example (non-exhaustive list) of a computer-readable medium includes: electrical connection section (electronic unit) with one or more cables, portable computer disk case (magnetic unit), RAM, ROM, EPROM/FM, optical fiber unit, and CD-ROM. In addition, the computer-readable medium may even be the paper or other suitable medium on which a program can be printed. The program can be obtained through optical scanning, editing, decoding or even by electronic processing for the paper or other mediums and stored in the computer memory.

It is understood that all parts of this invention can be implemented by hardware, software, firmware, or a combination of them. In the above execution modes, a plurality of steps or methods may be realized by the software or firmware stored in memory and executed by a suitable instruction execution system. For example, if realized by hardware as the another execution mode, any one of the following technologies disclosed in this field or their combination can be executed: discrete logic circuit with logic gate circuit for realizing logic function of data signal, special integrated circuit with suitable combination logic gate circuit, programmable gate array (PGA) and field programmable gate array (FPGA).

The common technicians of this field can understand that all or part of the steps

realizing the methods in the above embodiments can be completed by the hardware under the instructions of a program. The program can be stored in a computer-readable storage medium. When the program is executed, one or all steps of the method in embodiments can be included.

In addition, all functional units in each embodiment of this invention can be integrated into one processing module or be physically independent, or integrated into one module each two or more. The integration in the module can be realized by hardware or by functional module of software. If the post-integration module is realized by the functional module of software and sold or used as an independent product, it can be stored in a computer-readable storage medium.

The storage medium mentioned above can be ROM, disk or CD. Although the embodiments of this invention have been shown and described above, it can be understood that the above embodiments are exemplary and cannot be understood as the restrictions of this invention. The common technicians of this field can change, modify, replace and transform the embodiments above within the scope of this invention.

The above mentioned is only a preferred specific execution mode of this invention instead of the whole protection scope of this invention. Any change or substitution that a technician familiar with this technical field can get easily from the technical scope disclosed by this invention shall be covered by the protection scope of this invention. Therefore, the protection scope of this invention shall be subject to the protection scope of the claims.

CLAIMS:

1. A method of fingerprint extraction based on device of smart phone sensor, wherein the improvement in the step of:

Step I: acquiring of data;

Step II: pre-processing of the data;

Step III: recovering of ADC value;

Step IV: estimating gain matrix;

Step V: checking validity; and

Step VI: generating device fingerprint ID.

2. The method of fingerprint extraction according to Claim 1, wherein the improvement comprises the steps of step I above,

collecting data of gyroscope sensor, using $O = [O_0, O_1, \dots, O_{n-1}]$ to identify the data of single group number, and use $O_i = [O_x, O_y, O_z]^T$ to represent the acquisition of different axes.

3. The method of fingerprint extraction according to Claim 2, wherein the improvement comprises the step of step II, is featured,

The Step II includes:

Performing differential calculation on the data, after obtaining the data of the gyroscope sensor, by the equation as follows:

$$\Delta O = [O_1 - O_0, O_2 - O_1, \dots, O_{n-1} - O_{n-2}].$$

4. The method of fingerprint extraction according to Claim 3, wherein the improvement comprises in step II,

recovering the difference ΔA among consecutive ADCs and being estimated by the equation as follows:

$$\tilde{\Delta A} = \text{round}(H_0^{-1} \Delta O) \quad (1)$$

wherein, H^{-1} is the inverse matrix of the actual gain matrix H i.e. $H_0 = M_H \cdot I$, M_H is the nominal gain of the gyroscope, and I is the unit matrix;

eliminating the invalid values by the equation as follows:

$$E_{\Delta A}^r = \left| \tilde{\Delta A} - H_0^{-1} \Delta O \right| \quad (2) .$$

5. The method of fingerprint extraction according to Claim 4, wherein the improvement comprises in step iv, estimating the nominal gain $\tilde{H}$ by the equation as follows:

$$\tilde{H} = \Delta O / \Delta A .$$

6. The method of fingerprint extraction according to Claim 5, wherein the improvement comprises, in step v, quantifying the deviation between $\tilde{H}$ and true H, and calculating by the equation as follows:

$$E^e = \text{std}(\Delta O - \tilde{H} \Delta A)$$

wherein, std is the standard deviation function; and for error calculation:

$$E_H^r = \left| \tilde{H}^d \cdot 2^{16} - \text{round}(\tilde{H}^d \cdot 2^{16}) \right|$$

wherein, $\tilde{H}^d$ is the value of $\tilde{H}$ in dps

7. The method of fingerprint extraction according to Claim 1, wherein the improvement comprises in step VI generating fingerprint ID according to the formula as follows:

$$DeviceID = round(\tilde{H}^d \cdot 2^{16}) - round(H_0^d \cdot 2^{16})$$

8. The method of fingerprint extraction according to Claim 2, wherein the improvement comprises in step II, after obtaining the data of the gyroscope sensor, the obtained data is expressed as follows:

$$O = [O_0, O_1, \dots, O_{n-1}]$$

wherein, $O_i = [O_x, O_y, O_z]^T$ represents the uniaxial data, and the value of ΔO_{ix} is sorted and in ascending order, and $[\Delta O]_x$ represents the sorted array, likewise, the values of ΔO_{iy} and ΔO_{iz} , and output the results $[\Delta O]_y$ and $[\Delta O]_z$, and then calculate as follows:

$$\Delta \Delta O = [diff([\Delta O]_x) diff([\Delta O]_y) diff([\Delta O]_z)] \quad (3)$$

wherein, the formula of $diff(\Delta O)$ is:

$$diff(\Delta O) = [\Delta O_1 - \Delta O_0, \dots, \Delta O_{n-1} - \Delta O_{n-2}]$$

$$H^{-1} diff(\Delta O) = diff(\Delta A)$$

adding $\Delta \Delta O$ to ΔO , and performing this expansion repeatedly to generate more data with smaller values, generate multiple batches of data from the expansion data according to the range of values, and update the values; the absolute value of all elements shall be lower than the product of programming gains, that is:

$$\Delta O^{\varepsilon_i} = \{\Delta O_j \in \Delta O \mid \max(|\Delta O_j|) < (\varepsilon_i + 0.5)M_H\}$$

such batches are calculated, and then the values of each batch are doubled until is the values are the same as ΔO ; then calculate ΔO^{ε_i} step by step and update the value of H.

9. The method of fingerprint extraction according to Claim 8, wherein the improvement comprises in step III,

recovering the difference among consecutive ADCs and estimated by the equation as follows:

$$\tilde{\Delta A}^{ei} = \text{round}\left(\tilde{H}^{-1} \Delta O^{ei}\right) \quad (1)$$

wherein, $\tilde{H}^{-1}$ is the inverse matrix of the actual gain matrix H, i.e. $H_0 = M_H \cdot I$, M_H is the nominal gain of the gyroscope, and I is the unit matrix; and

eliminating the invalid values by the equation as follows:

$$E_{ei}^r = \left| \tilde{\Delta A}^{ei} - \tilde{H}^{-1} \Delta O^{ei} \right| \quad (2) .$$

10. The method of fingerprint extraction according to Claim 9, wherein the improvement comprises in step IV, after each update of $\tilde{H}$, checking whether $\Delta O^{ei} = \Delta O$ is tenable, if yes, pass the estimated $\tilde{H}$ to the validity check process, otherwise, return the updated $\tilde{H}$ to the pre-processing stage, and multiply the filter ordinal by 2 for the next batch of data processing.

11. An electronic device, comprising a memory, a processor and a computer program stored on the memory and capable of being operating by processor, characterized in that the processor runs the computer program in accordance with any of Claims 1-10.

12. A computer-readable storage medium with a computer program, characterized in that a processor runs the computer program in accordance with any of Claims 1-10.