i-Shield: A System to Protect the Security of Your Smartphone

Zhuolong Yu(✉), Liusheng Huang, Hansong Guo, and Hongli Xu

University of Science and Technology of China, Hefei, China
{yzl123,guohanso}@mail.ustc.edu.cn, {lshuang,xuhongli}@ustc.edu.cn

Abstract. Losing smartphones is a troublesome thing as smartphones are playing an important role in our daily lives. As smartwatches become popular, we argue that smartwatches can play a role in smartphone antitheft design. In this paper, we propose i-Shield, a real-time antitheft system that leverages accelerometers and gyroscopes of smartphones and smartwatches to prevent smartphone being stolen. As opposed to existing solutions which are based on Bluetooth, NFC, or GPS tracking, i-Shield follows a practical manner to achieve the goal of real-time antitheft for smartphones. i-Shield recognizes taken-out events of smartphones using a supervised classifier, and applies a dynamic time warping (DTW) scheme to recognize whether the events are caused by users themselves. We conduct a series of experiments on iPhone6 and iPhone4s, and the evaluation results show that our system can achieve 97.4 % true positive rate of recognizing taken-out actions, and classify taken-out actions with misclassification rate of 1.12 %.

Keywords: Antitheft · Smartphone · Smart wearable device · Supervised classification · Dynamic time warping

1 Introduction

Smartphones are now becoming very ubiquitous and extremely important in our daily lives. As smartphones always store lots of important personal information (even including credit card information), the security of smartphones is receiving more and more attention. A survey of smartphone theft victims conducted by IDG Research [3] shows that 1 in 10 U.S. smartphone owners are victims of phone theft, whose amount is 2.1 million in 2014. Smartphone producers have also brought out some solutions to smartphone theft [1,2,14], such as "Find My iPhone" of Apple and "Android Device Manager" of Android. Nonetheless, these approaches including password mechanism and fingerprint identification are effective only after the crime. Besides, Bluetooth technology is also employed for antitheft design, in order to detect the crimes when they are happening. But since Bluetooth can detect crimes only after criminals get about 6 m away (can be too late), it is not very effective in this scenario. Meanwhile, the majority of smartphones and smartwatches contain abundant advanced built-in sensors to sense users' motions. Sensor-based activity recognition has been

© Springer International Publishing AG 2016
F. Lehner and N. Fteimi (Eds.): KSEM 2016, LNAI 9983, pp. 453–464, 2016.
DOI: 10.1007/978-3-319-47650-6_36

studied extensively, especially for smartphones [12,13]. These researches focus on multiple fields, including health care [4,8,12], localization [11] and human computer interaction [6], but the field of smartphone antitheft has been paid little attention.

In this paper, we focus on the theft problem of smartphones. We propose i-Shield, a system employing sensors on smartphones and smartwatches to guard smartphones from being stolen. i-Shield can recognize taken-out actions of smartphones and check the sensor data of both smartphones and smartwatches to judge if the actions are caused by user themselves. The smartwatches will alarm users when i-Shield finds that the taken-out actions with high probability are not caused by users themselves.

The rest of this paper is organized as follows. In Sect. 2, we provide a system overview of our i-Shield system. Section 3 introduces the taken-out actions recognition part of i-Shield system, and Sect. 4 introduces how we judge whether the actions are secure. Section 5 reports the evaluation of our i-Shield system. We present the related works in Sect. 6, and give a discussion in Sect. 7.

2 System Overview

In this section, we give a brief introduction of i-Shield, our proposed smartphone antitheft system. Our system includes two parts which are based on smartphone and smartwatch respectively.

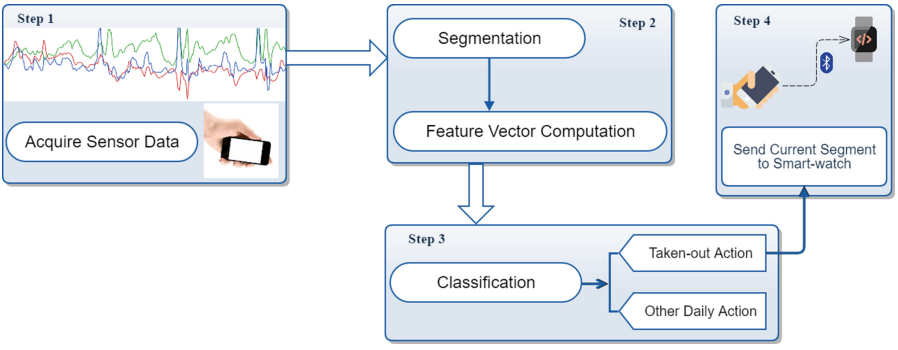

Fig. 1. System overview of i-Shield on smartphone side.

At the smartphone side, i-Shield firstly acquires the acceleration data and rotation-rate data from accelerometers and gyroscopes of smartphones, which are organized in the form of triples (x, y, z) with corresponding timestamps respectively. In the second step, we extract segments from the sensor data time series we obtain in the former step. Then, we compute a feature vector for each segment, which consists discriminative cues both in time-domain and frequency-domain. In the third step, i-Shield constructs a classifier on smartphone, it tells if there

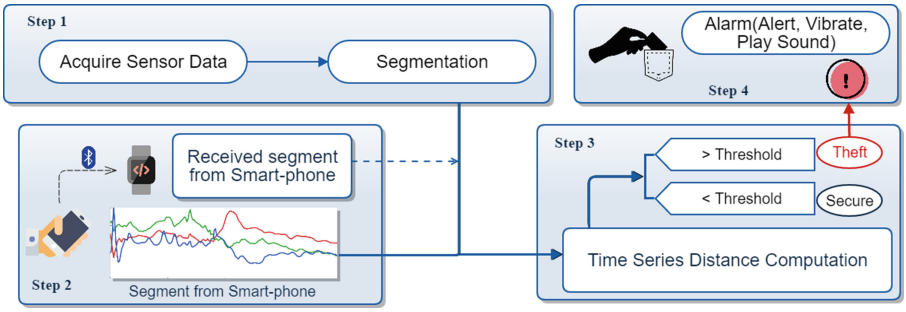

Fig. 2. System overview of i-Shield on smartwatch side.

is a Taken-out Action. When a Taken-out Action is recognized, smartphone side i-Shield will send current data segment to smartwatch side (Step 4) since there is a potential risk that smartphone is being stolen (Fig. 1).

Meanwhile, i-Shield does the same acquisition job and segmentation job at smartwatch side. Right after receiving data segment from smartphone side, i-Shield computes distance between two series of sensor data sequences from smartphone and smartwatch. Based on the distance measured, i-Shield judges if the taken-out actions are with high probability caused by users themselves. At last, i-Shield will arouse an alarm event (alert, vibrate, or play sound) when a risky action is detected. The user can choose to turn the alarm off manually (Fig. 2).

3 Recognize Taken-Out Actions

In this section, we specifically describe the recognition process i-Shield implement on smartphone side. The recognition process mainly needs three steps to recognize an action, which are Segmentation, Feature Vector Computation and Classification respectively. Firstly, i-Shield divides sensor data time series into segments, then feature vectors are computed based on segments. After feature vectors computed, classification algorithms are applied, segments can be classified into specific actions.

3.1 Segmentation

To detect taken-out actions precisely and efficiently, sensor data time series are divided into segments of appropriate length.

Figure 3 plots the waveform of accelerometer data generated by 6 min of daily activities including walking, going upstairs and downstairs, and having short rest. As shown in Fig. 3, we extract two segments from the original data. Segment A shows a intense wave (actually, taking out phone while walking), while Segment B shows a flat wave (having a short rest).

In this paper, we focus on potentially dangerous actions, namely Taken-out Actions, which will cause waveforms of a certain shape, e.g. waveform of

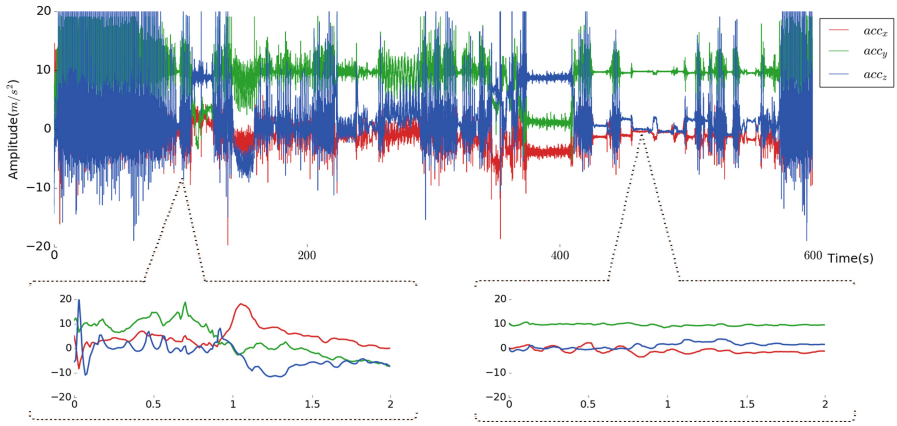

Fig. 3. Waveform of accelerometer data.

Segment A. In the meantime, Segment B indicates an almost dormant state that we regard as a safe action. Accordingly, we do not need to purchase all available segments, but only the segments containing waveforms at least of a certain intensity.

Sliding Windows [15] and End-Points Detection [7] are two of the most popular segmentation methods. As Sliding Window-based segmentation algorithm will collect massive segments including ones we will not need, and cause a enormous consumption of computing resources, memories and energy. We apply End-Point Detection-based algorithm in our system, since it can extract specific segments we are interested in.

3.2 Feature Vector Computation

For later use of classification, we compute a set of features for each segment, which are organized as a feature vector. In an attentive manner, we choose 24 features for each segment, which can be divided into three categories: Time Domain Features, Frequency Domain Features, and Statistics Features respectively. These features are listed below in Table 1 (For simplicity, we use Acc. to denote acceleration value of 3 axes, and Rtr. to denote rotation-rate value).

- **Time Domain Features.** Time Domain Features are intuitional and they can be obtained with low computational complexity.

  We extract minimum, maximum and mean value of acceleration data and rotation-rate data on 3 axes. These values can describe approximate shape of sampled data, and have been exploited generally.
- **Frequency Domain Features.** In order to describe periodic characteristics of sampled data, we leverage Fast Fourier Transform (FFT) to transform our time series segment into frequency domain.

  We extract **Peak Amplitude**, **Peak Frequency** and **Spectral Slope** of Acc.Z.

Table 1. Feature set for each sensor data segment

Time Domain Features	
Accelerometer:	$min(Acc.X)$, $max(Acc.X)$, $mean(Abs(Acc.X))$, $min(Acc.Y)$, $max(Acc.Y)$, $mean(Abs(Acc.Y))$, $min(Acc.Z)$, $max(Acc.Z)$, $mean(Abs(Acc.Z))$, $mean(\sqrt{Acc.X^2 + Acc.Y^2 + Acc.Z^2})$
Gyroscope:	$min(Rtr.X)$, $max(Rtr.X)$, $mean(Abs(Rtr.X))$, $min(Rtr.Y)$, $max(Rtr.Y)$, $mean(Abs(Rtr.Y))$, $min(Rtr.Z)$, $max(Rtr.Z)$, $mean(Abs(Rtr.Z))$, $mean(\sqrt{Rtr.X^2 + Rtr.Y^2 + Rtr.Y^2})$
Frequency Domain Features	
Accelerometer:	$PeakAmplitude(Acc.Z)$, $PeakFrequency(Acc.Z)$, $SpectralSlope(Acc.Z)$
Statistics Features	
Accelerometer:	$Kurtosis(Acc.Z)$

– **Statistics Features.** We calculate **Kurtosis** of acceleration value on Z-axis. This feature weighs how the amplitude decays near the extreme points, namely the peakedness and flatness. The Kurtosis of accelerometer value on Z-axis is calculated as:

$$Kurtosis_i = \frac{n \sum_{j=1}^{n} (z_j - \bar{z})^4}{(\sum_{j=1}^{n} (z_j - \bar{z})^2)^2}$$

where z_j indicates the value of the j-th sampling point in segment $Acc.Z$ and $\bar{z}$ indicates the mean value of all sampling points in segment $Acc.Z$. n is the length of segment $Acc.Z$.

3.3 Classification

We divide all actions into three categories: taken-out when user is still, taken-out when user is walking, other daily activities. We mark the three categories as TOS, TOW, and OTHER respectively. Note that OTHER includes all actions of daily activities except for taking smartphone out of pocket, such as walking, jogging, riding and so on (Table 2).

We construct four classifiers based on Hoeffding Tree, Logistic, Naive Bayes, and Random Forest respectively. The result of classification is shown in Sect. 5.1. When an action is recognized as a Taken-out Action (TOS or TOW), i-Shield will send the current segment to smartwatch right away through Bluetooth.

4 Check If Actions Are Secure

In this section, we describe how i-Shield works at smartwatch side. i-Shield will continuously acquire acceleration and rotation-rate data of smartwatch, and

Table 2. Three categories of actions.

Catagory	Actions
TOS	taken-out when user is still
TOW	taken-out when user is walking
OTHER	still, walking, jogging, riding, sitting down, standing up, going upstairs and downstairs, working out at the gym, etc. (phone in hand and phone in pocket respectively)

keep a constant length log using a circular queue. When a data segment comes from smartphone through Bluetooth, the time-series distance computation procedure will be activated. Depending on the distance of time-series, i-Shield judges whether the action is caused by user, if yes, the action is secure. At last, i-Shield will alarm if the action is insecure. We apply dynamic time warping (DTW) algorithm to accomplish the task of time-series distance computation.

Dynamic time warping (DTW) has been wildly used in the field of speech recognition, signature recognition, shape matching and etc. DTW measures similarity between two temporal sequences that may vary in time or speed. By applying DTW, we don't need to worry difference in both absolute-time (always different between devices) and sample-rate between smartwatches and smartphones.

Given two time series $S = [s_1, s_2, ..., s_n]$ and $T = [t_1, t_2, ..., t_m]$, let $Dist[i, j]$ denotes the distance between symbol s_i and t_j, that is

$$Dist[i, j] = (s_1.X - t_1.X)^2 + (s_2.Y - t_2.Y)^2 + (s_3.Z - t_3.Z)^2 \quad (1)$$

Following DTW, we define F[i, j] which satisfies:

$$F[i, j] = Dist[i, j] + minimum(F[i - 1, j - 1], F[i, j - 1], F[i - 1, j]) \quad (2)$$

Finally, we judge if the actions are caused by users themselves in terms of F[n, m] of acceleration and rotation-rate. When F[n, m] is below the threshold line, we judge that action is caused by user safely. The detail figure is shown in Figs. 7 and 8 in Sect. 5.2.

Fig. 4. Coordinate system of smartwatch and smartphone are opposite.

Note that in most cases, coordinate system of smartwatch and smartphone will not be the same. We consider the most natural way shows in Fig. 4, that the

screens of watches and phones should be facing users (not hands or wrists), top and bottom should not be upside down. In i-Shield, we conform smartwatch's X, Y, Z-axis value (acceleration and rotation-rate) to smartphone's coordinate system. We denote before-conform X, Y, Z-axis value of smartwatch as (x, y, z), and after-conform value as $(\tilde{x}, \tilde{y}, \tilde{z})$, we have:

$$\left(\tilde{x}, \tilde{y}, \tilde{z}\right) = \left(x, y, z\right) \begin{pmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{3}$$

5 Evaluation

In order to ease the development process, we make use of a smartphone tied to a user's wrist as a substitute for a smartwatch. We packed i-Shield's functionality of smartphone side and smartwatch side together as an application, and implemented our system on iOS9.

We evaluate our proposed system using an iPhone6 Plus as our to-be-protected smartphone, an iPhone4s (small enough to be tied on a wrist) as our smartwatch simulator. Our evaluation has three parts. We start with evaluating how well we recognize taken-out actions on smartphones. Then, we report the result of smartwatch side actions checking. Finally, a real world evaluation is presented.

5.1 Recognizing Taken-out Actions

Our dataset consists of 511 TOS actions, 391 TOW actions and 35756 OTHER segments, collected by 6 volunteers for over 2 days. We construct six different classifiers, respectively based on Hoeffding Tree, Logistic, Naive Bayes, Random Forest, k-Nearest Neighbors, and Multilayer Perceptron. We conduct a series of 10-fold cross-validation experiments, and the results are shown in Table. 3, Figs. 5 and 6. Table. 3 presents the confusion matrixes of the six classifiers. Figure 5 illustrates two histograms of true positive rate (TPR) and false positive rate (FPR). Figure 6 reports time consumption of the six classifiers on our dataset. Table. 3 and Fig. 5 indicates that, k-Nearest Neighbors achieves the largest true positive rate of taken-out actions (TOS and TOW) which guarantees reliable safety for smartphones. While Random Forest will barely cause a false alarm on smartwatch as it outputs only 3 times of false positive recognition of taken-out actions which is the least among the six classifiers. In the meanwhile, the true positive rate of Random Forest to recognize taken-out actions is as high as 97.4 %. As presented in Fig. 6, Multilayer Perceptron and k-Nearest Neighbors both consume much more time than other four classifiers. Even though k-Nearest Neighbors obtains the best accuracy on recognizing taken-out actions, its total time consumption is 104.95 s that is 12 times as large as Random Forest consumes (8.78 s). Note that Random Forest consumes most of its total time on training part (7.97 s), while test time is only 0.81 s on our dataset (36658 items in total).

Table 3. Time consumption and Confusion matrixes of six classifiers. Note that a, b, c stands for TOS, TOW and OTHER respectively.

	Hoeffding Tree			Logistic			Naive Bayes		
Classified as:	a	b	c	a	b	c	a	b	c
a	**478**	**0**	**33**	**494**	**2**	**15**	**478**	**0**	**33**
b	**6**	**380**	**5**	**23**	**361**	**7**	**7**	**382**	**2**
c	**32**	**1**	**35723**	**26**	**23**	**35707**	**32**	**1**	**35723**
	Random Forest			**KN-Neighbors**			**ML-Perceptron**		
Classified as:	a	b	c	a	b	c	a	b	c
a	**490**	**0**	**21**	**509**	**0**	**2**	**480**	**0**	**31**
b	**1**	**388**	**2**	**3**	**388**	**0**	**0**	**391**	**0**
c	**0**	**3**	**35753**	**6**	**1**	**35749**	**4**	**1**	**35751**

Since our system runs training process only once during system initialization, Random Forest is high-efficient in our system. So we implement Random Forest in our proposed system, it achieves approximately 0.0 % of false positive rate of recognizing taken-out actions, and the true positive rate is 97.4 %.

5.2 Check If Actions Are Secure

In this section, we present the evaluation of i-Shield on smartwatch side. We collected 100 time-series pairs of phones and watches which generated when phones were taken out by users themselves, and 200 time-series pairs generated when phones were taken out by others. Here we define the former 100 time-series pairs as Safe Actions, the latter 200 time-series pairs as Unsafe Actions. We obtained distances of acceleration & rotation-rate for both Safe Actions and Unsafe Actions using DTW mentioned in Sect. 4. Figure 7 shows the distribution of Safe Actions and Unsafe Actions in terms of the distances we got.

As shown in Fig. 7, Safe Actions aggregate at bottom-left corner, which means Safe Actions cause very small-scale distances compared to Unsafe Actions. We

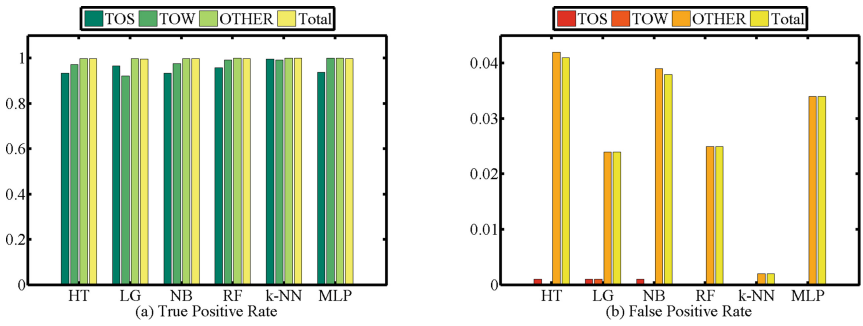

Fig. 5. TP rate (left) and FP rate (right) of six classifiers.

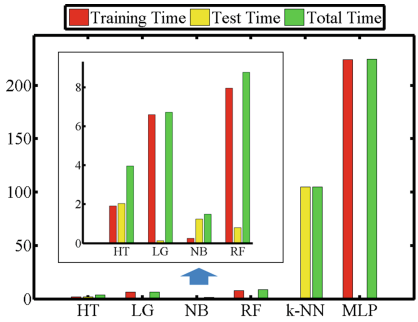

Fig. 6. Time consumption of six classifiers.

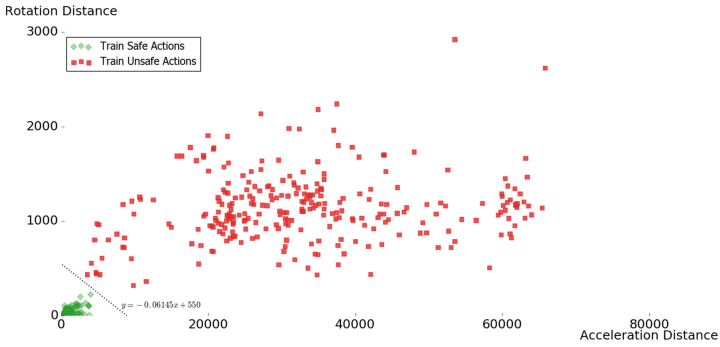

Fig. 7. Distribution of Safe Actions and Unsafe Actions of training set.

graph line $y = -0.06145x + 550$ using a linear classifier. Time-series pairs whose distances appear below $y = -0.06145x + 550$ are considered as Safe Actions, others are considered as Unsafe Actions.

We then calculated distances of other 178 Safe Actions and 424 Unsafe Actions using DTW as a test set and plot the result in Fig. 8. It's shown that only two of the Safe Actions are recognized as Unsafe Actions with misclassification rate of 1.12 %, while no Unsafe Actions are misclassified. Commonly speaking, there may be false alarms with very small probability, but will seldom be any theft situations regarded as safe situations.

5.3 Real World Evaluation

To study how well would i-Shield work in real world situation, we conducted a real world evaluation with eight volunteers. They are all smartphone users, and five of them are smartwatch users.

In the study, we let four of them act as users, carrying smartphone in the pocket and "smartwatch" on the wrist, while other four act as thieves trying to steal users' smartphones. We conducted 120 theft actions, 80 safe taken-out

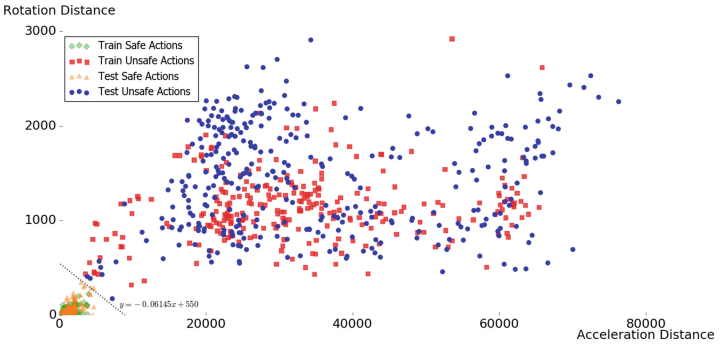

Fig. 8. Distribution of Safe Actions and Unsafe Actions.

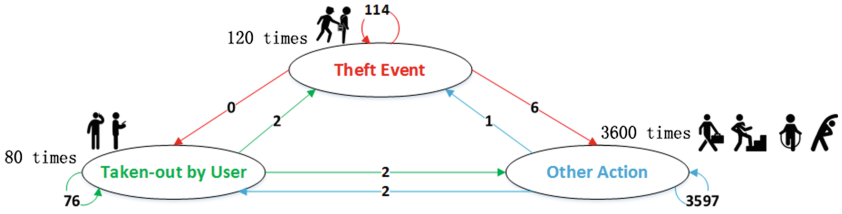

Fig. 9. Result of real world evaluation.

actions and 3600 other daily action segments among these four user-thief pairs. As shown in Fig. 9, 114 of theft actions were recognized correctly with other 6 are missed by i-Shield. False alarm appeared 3 times as two safe actions and one other daily action were recognized as theft action.

After the study, a volunteer who had once been a victim of smartphone theft said, "This system works so sensitively and naturally, it's just like an invisible shield of my cell phone!"

6 Related Work

Smartphone related works have been studied extensively in recent years. Great progress has been made in various fields about smartphone including but not limited to motion detection [4,12,13], location based service [11], security [10], energy consumption saving [5]. In this section, we describe two areas of related work.

Motion sensor related works. Motion sensor has been studied a lot, especially after smartphones appear. As for smartphones, Thompson et al. [12] proposed a system detecting car accidents using sensors of smartphones. Abbate et al. [4] proposed a smartphone-based fall detection system with concern for health of elderly people. As for smartwatches, Gouthaman et al. [6] designed a system to control computers using accelerometer, gyroscope and gravitational

sensors of smartwatches. Parate et al. [8] proposed a trajectory-based method that extracts hand-to-mouth gestures to recognize smoking gestures. In addition, various commercial applications based on motion detection are available on both smartphones and smartwatches, and provide helpful and delightful functionalities, such as step counting and sleep tracking. However, motion detection hasn't been applied to the field of smartphone antitheft.

Mobile phone antitheft related works. Mobile phone antitheft design has been studied for many years, since mobile phone theft becomes a serious problem. Whitehead et al. [14] gave a review of mobile phone antitheft designs. They reviewed a great many antitheft designs, but most of them are traditional ways, which are not very practical nowadays. Ren et al. [9] proposed a model that PC and smartphone would form a loop to track each other. Yu et al. [16] leveraged emergency call mechanism to achieve remote deletion on stolen phones. These models works only after our devices are missing, in which condition the devices are probably turned off by criminals and not trackable.

7 Discussion

We plan to extend our current system to support other situations with more comprehensive functions.

Location based. We are planning to take location information into consideration to make our i-Shield system more intelligent. For example, crowded places like subway stations can be considered as high-risky places, i-Shield needs to lower the threshold line (be more sensitive), to guarantee invulnerably safety. Likewise, i-Shield needs to lower its sampling rate to reduce energy consumption when at home.

Smart wallets. Our work can be extended to the field of smart wearable devices. Wallets which are as popular as smartphones in crime of theft, can be protected like the way smartphones are, after embedded with sensor-based computation chips.

Acknowledgements. This paper is supported by the National Science Foundation of China under No. U1301256 and 51274202, Special Project on IoT of China NDRC (2012-2766).

References

1. Apple. http://www.apple.com/icloud/find-my-iphone.html
2. Google. https://www.google.com/android/devicemanager
3. Lookout. https://www.lookout.com/resources/reports
4. Abbate, S., Avvenuti, M., Bonatesta, F., Cola, G., Corsini, P., Vecchio, A.: A smartphone-based fall detection system. Pervasive Mob. Comput. **8**(6), 883–899 (2012)

5. Chen, X., Jindal, A., Ding, N., Hu, Y.C., Gupta, M., Vannithamby, R.: Smartphone background activities in the wild: origin, energy drain, and optimization. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, pp. 40–52. ACM (2015)
6. Gouthaman, S., Pandya, A., Karande, O., Kalbande, D.: Gesture detection system using smart watch based motion sensors. In: 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), pp. 311–316. IEEE (2014)
7. Gu, T., Chen, S., Tao, X., Lu, J.: An unsupervised approach to activity recognition and segmentation based on object-use fingerprints. Data Knowl. Eng. **69**(6), 533–544 (2010)
8. Parate, A., Chiu, M.C., Chadowitz, C., Ganesan, D., Kalogerakis, E.: RisQ: recognizing smoking gestures with inertial sensors on a wristband. In: Proceedings of the 12th annual international conference on Mobile systems, applications, and services, pp. 149–161. ACM (2014)
9. Ren, B., Sun, Y., Lin, Y.: Anti-theft and tracking loop model based on PC and smart phone. In: 2013 Fifth International Conference on Computational and Information Sciences (ICCIS), pp. 1943–1946. IEEE (2013)
10. Shao, J., Lu, R., Lin, X.: Fine: a fine-grained privacy-preserving location-based service framework for mobile devices. In: IEEE Conference on Computer Communications, IEEE INFOCOM 2014, pp. 244–252. IEEE (2014)
11. Shu, Y., Shin, K.G., He, T., Chen, J.: Last-mile navigation using smartphones. In: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking. pp. 512–524. ACM (2015)
12. Thompson, C., White, J., Dougherty, B., Albright, A., Schmidt, D.C.: Using smartphones to detect car accidents and provide situational awareness to emergency responders. In: Cai, Y., Magedanz, T., Li, M., Xia, J., Giannelli, C. (eds.) MOBILWARE 2010. LNICST, vol. 48, pp. 29–42. Springer, Heidelberg (2010). doi:10.1007/978-3-642-17758-3_3
13. Weiss, G.M., Timko, J.L., Gallagher, C.M., Yoneda, K., Schreiber, A.J.: Smartwatch-based activity recognition: a machine learning approach. In: 2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), pp. 426–429. IEEE (2016)
14. Whitehead, S., Mailley, J., Storer, I., McCardle, J., Torrens, G., Farrell, G.: In safe hands: a review of mobile phone anti-theft designs. Eur. J. Crim. Policy Res. **14**(1), 39–60 (2008)
15. Wu, W., Dasgupta, S., Ramirez, E.E., Peterson, C., Norman, G.J.: Classification accuracies of physical activities using smartphone motion sensors. J. Med. Internet Res. **14**(5), e130 (2012)
16. Yu, X., Wang, Z., Sun, K., Zhu, W.T., Gao, N., Jing, J.: Remotely wiping sensitive data on stolen smartphones. In: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security, pp. 537–542. ACM (2014)