2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)

SmartDog: Real-time Detection of Smartphone Theft

Shan Chang, Ting Lu, Hui Song
School of Computer Science and Technology
Donghua University, Shanghai, China
{changshan, luting, songhui}@dhu.edu.cn

Abstract— With the enhanced computing capabilities and a wide variety of functions available on smartphones, critical and sensitive information, such as contact lists, messages, schedules, credit card numbers, is stored on smartphones, which makes preventing smartphone from being stolen of an unprecedented importance. Loss of smartphones not only causes economic loss but also jeopardizes the privacy of the owners. Existing anti-theft schemes only provide passive protection, which remotely manipulates lost phones to lock and delete private information. However, if the stealer deletes the protection apps or shuts down the phone before the owner being aware that the phone has been stolen, remedial actions become invalid. Hence it is of great importance to detect smartphone theft as soon as it happens, such that stealing can be stopped. In this paper, we propose SmartDog, a real-time smartphone anti-theft scheme for keeping smartphone safe. With embedded motion sensors, SmartDog can effectively capture the unique and reliable biometrical features of the owners about how they pick up the smartphone from their pockets or bags. If a stealer tries to steal a smartphone away from the pocket or bag of the owner, SmartDog will detect the unusual motion, even if an attacker can see an owner picking up his/her phone (Since the attacker can hardly reproduce the same behavior). We implement SmartDog and conduct intensive trace-driven simulations with 'picking up' samples from 20 volunteers, collected over two weeks. SmartDog achieves 10.2% average false positive and 5.5% average false negative error rates, respectively.

Keywords—smartphone theft detection, real-time, inertial sensors, active protection

I. INTRODUCTION

A report from the global market investigation institution TrendForce [1] shows that the volume of global smartphone shipment was 1.293 billion in 2015, which has an increase of 10.3% comparing with 2014. It is expected that the global shipment can reach to 1.5 billion in 2016. With the enhanced computing capabilities, smartphones are no long simple communication tools, and more and more programs running on the desktop terminal have been migrated to smartphones, enabling rich functionalities. Smartphones have become essential to daily life of people. Users get used to store personal information on their smartphones, such as mailing lists, pictures, videos, schedules, and even bank accounts and passwords. Thus once smartphones are stolen and fall into the hands of malicious ones, critical private information might be leaked, causing serious damage. For example, a stealer might use a stolen phone to withdraw cash, or to defraud the relatives and friends of the victim. A report of BBC news [2] mentioned: "314 mobile phones are stolen on London's streets every day, according to the Metropolitan Police", and "Met statistics showed 56,680 mobiles - 28,800 of those iPhones - were reported stolen in London between April and September last year". Not only in Britain, but this has become global problem, and the number of phone thefts has also shown a rising trend. In Jan. 2011, Symantec Company, focusing on the personal information protection, has conducted a survey on the mobile phone security problem (Norton Cybercrime Report 2011 [3]), which includes 500 people who are 18 to 54 years old. The result shows that 74% Chinese mobile phone users have suffered phone lost or stolen. Symantec also carried out a Smartphone Honey Stick Project involving 60 "lost" smartphones [4]. Those smartphones are dropped in six different cities in Canada. They were left in high traffic public places, such as elevators, malls, food courts, and public transit stops. Then project members waited to see what would happen. The result shows that 93% of the phones were accessed by finders for searching information on the device, with 63% of the phones being accessed for corporate apps or data and 83% being accessed for personal apps or data. Symantec released "Android mobile phone security survey" in 2013. The survey shows that over 80% of the victims feel very anxious about phone lost, since it incurs privacy leakage, and recovery the lost information is not easy. Therefore, anti-theft tools should be very useful and have a vast application prospect.

There is a rich set of anti-theft schemes in both the industry and the literature. Existing anti-theft schemes only provide passive protection, which remotely manipulates lost phones to lock or delete private information. Anti-theft apps, like Tencent Mobile Phone Manager [5], Rising Phone Security Assistant [6] etc., are based on SIM card replacement detection. These apps automatically record the new SIM card number, and then the owner can utilize Short Message Service (SMS) to send backing up, deleting, and locking commands remotely. X. Yu et al. propose a remote deletion mechanism, allowing the phone owner to delete the private data remotely even if Wi-Fi is disabled and the SIM card is unplugged [7]. L. Subramanian et al., propose an architecture for providing security services in the cloud for smartphones within a corporate environment [8]. However, if the stealer deletes the anti-theft apps or shuts down the phone before the owner being aware that the phone has been stolen, those remedial actions become invalid.

In this paper, we propose an active protection scheme, called SmartDog, enabling real-time smartphone theft detection. SmartDog is based on the 'picking up' features of smartphone owners. One 'picking up' motion refers to an action of an owner taking the smartphone out of his or her pocket or bag. In

978-1-5090-5880-8/16 $31.00 © 2016 IEEE
DOI 10.1109/iThings-GreenCom-CPSCom-SmartData.2016.61

essence, SmartDog adopts a machine learning methodology, consisting of a training phase and a detection phase. More specifically, in the training phase, SmartDog first asks a legitimate user to collect a small number of 'picking up' samples. Several weak classifiers are derived from the raw readings of the embedded 3D accelerometer and gyroscope sensors, to distinguish legal picking up motions of an owner from abnormal motions of stealers. Obtained weak classifiers are further integrated to train a strong classifier, using an AdaBoost algorithm, to improve the accuracy of SmartDog. In the detection phase, SmartDog uses the pre-trained strong classifier to verify the legitimacy of picking up attempts of a user, and unlock the phone if the user passes the verification (since we consider that the purpose of owner picking up the phone is to use it). Otherwise, SmartDog will treat the attempts as unusual events, which implies that stealing is taking place. Consequently, an alarm is triggered immediately. The key insight behind SmartDog is that people have consistent and distinguishing physiological characteristics (e.g., the physical structure of the arm) and behavioral characteristics (e.g., picking up behavior patterns).

Comparing with the passive protection schemes, SmartDog has the following novelties.

1. SmartDog actively detects thefts as long as stealers are trying to steal smartphones away from the pockets or bags of the owners. Thus SmartDog prevents smartphones form being stolen rather than conducts remedial actions after thefts.
2. SmartDog does not rely on any types of communication technologies, such as Wi-Fi, SMS, Cellular etc., which makes SmartDog more robust and reliable.
3. SmartDog is difficult to compromise as it is very hard for a stealer to generate the same picking up motion as the owner does, through shoulder surfing or biometrics hacking attacks.
4. SmartDog is quite reliable and works well with various modes of transport such as buses and subway trains, escalators, and in different user postures, sitting and standing.
5. As accelerometers and gyroscopes are widely available sensors in most off-the-shelf smartphones, it is easy to deploy SmartDog.

We implement and evaluate the performance of SmartDog via trace-driven simulations with picking up samples collected from 20 volunteers over two weeks. The results show that the average false positive and false negative error rates of SmartDog are 10.2% and 5.5%, respectively.

The remainder of this paper is organized as follows. Section II introduces related work. Section III describes system model. We present the data collection and pre-processing in Section IV. Section V introduces the design of SmartDog. Section VI presents the performance evaluation and experiment results. Finally, we present concluding remarks of our work and summarize the directions for future work in Section VII.

II. Related Work

At present, most smartphone anti-theft schemes only provide passive protection, which aims to reduce privacy leakage and information loss after smartphones being stolen. When such emergency happens, the victim sends control messages to the stolen phone, in order to lock and localize the phone, backup and delete data, remotely, which are nothing to do with reducing the risk of phone lost. Synchronica Plc developed a software, named Mobile Manager, to support Symbian smartphones. With Mobile Manager, businesses or service providers can immediately secure lost or stolen devices remotely, from Mobile Manager's web-based application. The device can be wiped and locked over-the-air, preventing sensitive corporate or personal data stored on the device from being accessed by unauthorized users [11]. Tencent Mobile Phone Manager [5], Lookout Mobile Security [12], Rising Phone Security Assistant [6] etc. are based on SIM card detection. Once the SIM card is replaced, the system will automatically record the new SIM card number, and then the owner can utilize SMS to send backing up, deleting, and locking commands remotely. Other apps, such as 360 [13] and Kingsoft [14] Mobile Phone Guard, provide SMS notification for SIM card replacement, phone tracking, alarming, and locking. X. Yu et al. propose a remote deletion mechanism that allows the phone owner to delete the private data remotely even if Wi-Fi is disabled and the SIM card is unplugged [7]. A. U. S. Khan et al. present a technique to enable anti-theft for android based mobile phones by using services like MMS (Multimedia Messaging Services) instead of SMS [9]. L. Subramanian et al., propose an architecture for providing security services in the cloud for smartphones within a corporate environment [8].

However, those passive anti-theft methods assume that the victim discovers the lost of his or her smartphone immediately, and manipulates the phone remotely before the thief shutdown the phone or remove the protection application service from it. Even if the phone has set the keyboard lock protection program, thieves can also remove the protection through rooting the system. L. Simon et al. study the "anti-theft" mechanisms available to consumers to thwart unauthorized access to personal data on stolen Android smartphones [10]. They investigate the implementation of their "remote wipe" and "remote lock" functions on 10 popular anti-theft apps. They found that remote locks are unreliable due to poor implementation practices. In summary, current anti-theft tools cannot prevent phones from being stolen, and can be disabled through various approaches. It is necessary to design an active and real-time anti-theft scheme, which can detect the theft, and alert the victim in the first place, preventing stolen from happening.

III. System Model

In the system of real-time detection of phone theft, we consider the following three key entities:

- Smartphones: are the devices to be protected. We require such a target smartphone to have an onboard accelerometer, a gyroscope, and a digital compass, which can constantly measure the motion and attitude of the smartphone, respectively. We have very limited requirements on the computation and storage capabilities of the smartphone and rely on no other special hardware.
- Smartphone Owners: have the right to access a smartphone. In order to reduce the power consumption of smartphones, we require the owner to help deciding when SmartDog should monitor the smartphone. We require that owners should put their phone in their pockets or single shoulder bag, while do not put their smartphones in double shoulder bags, where is hard for the owner to pick up the phone, and easy for stealers to steal it. Note that as we do not require an owner to pick up his or her smartphone with any special manners, owners can choose their preferred or habitual ways.
- Stealers: are deliberate attackers who try to steal smartphones away from the pockets or bags of the owners. We assume stealers cannot have physical access to a smartphone during the training phase of SmartDog. After the training phase, stealers have the following three capabilities. First, they can get close to smartphone owners in public places such as metro, elevator, bus and street, and then snatch a phone without being noticed. Second, stealers can launch shoulder surfing attacks by spying or even recording the owner when he/she brings out the phone from the pocket or bag. Third, stealers have necessary equipment and technologies to launch biometrics hacking attacks.

IV. DATA COLLECTION

In this section, we describe the process for collecting and pre-processing raw "picking up" data from smartphones.

A. Collecting "picking up" Data

We collect "picking up" data with Samsung Galaxy Note 3, a standard Android smartphone, with which raw readings on each axis of the 3D accelerometer and the gyroscope embedded on the phone can be recorded. The sampling frequency is 200Hz. We recruit 20 volunteers, five females and fifteen males, aged from 18 to 34, including three undergraduate students, thirteen graduate students, and four faculty members. We collect data in four scenarios: shopping mall, bus, metro and escalator. In each scenario, volunteers help collect their "picking up" data in two postures, i.e., sitting and standing, since we consider that smartphones are most likely to be stolen when owners remain stationary. Under each posture, smartphones are located in three places: jacket pocket, pants pocket, and shoulder bag. We also design a set of experiments to imitate stealers who abstract smartphones from pockets or bags of the owners in scenarios as mentioned above. Under each configuration, each volunteer was asked to pick up the phone for fifty times within two weeks. We collect the picking up data set in the year of 2014, i.e., two weeks from Mar. 10 to Mar. 23.

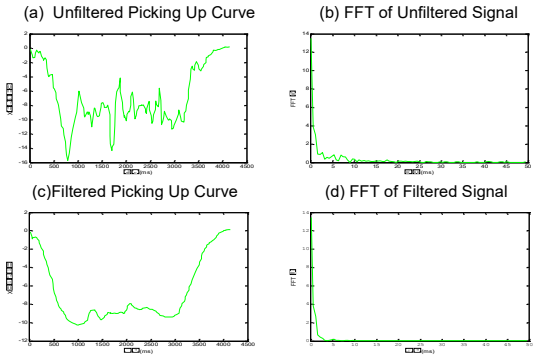

Figure 1. Unfiltered and filtered picking up signals of the x-axis of acceleration.

B. Removing Noise

The time series of acceleration and angular speed along each axis can be treated as signals, which contains high frequency noise. This can be seen in Figure 1(a), which shows the x-axis acceleration readings of a picking up motion. We consider frequencies above 10Hz as noise because it can be seen that most energy is contained in frequencies below 10Hz as we can see from the Fast Fourier transform (FFT) result shown in Figure 1(b). We need to remove such high frequency noise, as it would affect the results of picking up velocity and angular speed calculation.

V. REAL-TIME DETECTION OF PHONE THEFT

A. Overview of SmartDog

In general, SmartDog consists of two phases: training phase and detection phase. In the training phase, picking up motion of a smartphone owner is first captured by a motion estimator, which outputs the raw three-dimension acceleration and angular speed signals. Then above signals are provided to a DTW-based algorithm for weak classifier learning. With those motion signals, six weak classifiers are produced, and each of them is trained using single dimension signal of an inertial sensor. Then, obtained weak classifiers are further integrated to train a strong classifier using AdaBoost algorithm. In the detection phase, we consider two cases: the picking up motion from the owners and others (stealers). Once a person takes a phone out from the owner's pocket or bag, the motion is sent to SmartDog to determine the legitimacy of the person based on pre-trained strong classifier. The architecture of SmartDog is shown in Figure 2.

B. Motion Estimator

Motion estimator performs two key functions. First, it determines whether or not a person is taking the phone out of the owner's pocket or bag. Second, if 'picking up' is detected, the raw sensor readings of accelerometer and gyroscope will be provided to weak classifier learning module. Considering the real-time requirement of detection, we adapt a simple way to detect picking up motion: if the accumulative movement of the

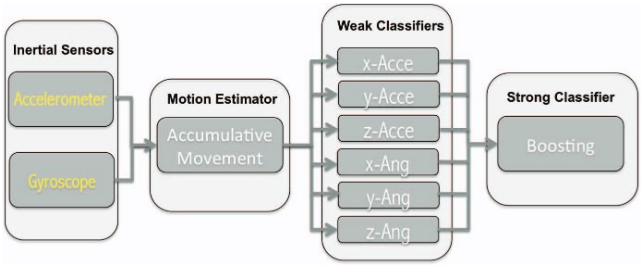

Figure 2. Architecture of SmartDog. Arrow lines show the data flow.

phone within a small period of time (2 seconds) is larger than a pre-decided threshold, then the estimator provides the sensor readings and trigger a control signal to enable classifiers. In practice, we choose a small threshold in order to reduce false negative errors.

C. DTW-based Weak Classifier

Dynamic Time Warping (DTW) is an algorithm for measuring similarity between two temporal sequences which may vary in speed. DTW is a kind of nonlinear technology which combines time warping with distance measurement. The output of the algorithm is the minimum distance of two temporal sequences. Small distance implies high similarity. DTW can reduce the influence of distortion and shifting in temporal sequences, which has been widely applied on temporal sequences of video, audio and graphics data. Comparing with Support Vector Machine (SVM), Bayesian Network and Decision Tree, DTW is less affected by the continuity of the sampled data, hence is more suitable for processing the time series with distortion.

In DTW, Warp path is a matching sequence of data points on temporal sequence X to those on Y, denoting as , and . The subscript k of represents , which are the indexes of a pair of data points in X and Y. and refers to and , respectively, which ensures that every data point in X and Y will appear in the warp path.

Warp Path Distance is the total distance of data pairs on the warp path, representing as:

$$\sum_{k}$$

we calculate using Euclidean distance, where denotes the i-th point of X, and denotes the j-th point of Y. The Minimum Warp Path Distance (MWPD) is the output of DTW algorithm, denoted as , used to measure the similarly of time series X and Y.

SmartDog adapts DTW to measure how close can two picking up time sequences be. In training procedure, suppose that there are training time sequences from an owner of a smartphone, denoted as . In time sequence (which represents i-th picking up motion of the owner), j-th sample point is a 6-dimension vector, which contains the values of acceleration and angular speed from three orthogonal directions, i.e., axis, axis, and axis, representing as . For k-th dimension ( ), we use the open source implementation of DTW to calculate the minimum warp path distance between any two different sequences and in T. The maximum value of is selected as the threshold of the corresponding dimension , deciding if two sequences are similar or not, i.e.,

Table 1: The DWT-based algorithm for classifier learning

Algorithm 1: DTW-based Classifier Learning

Input: Given time sequences the j-th sample point in is

For

Calculate

Choose , minimizing

Output: The final classifiers are:

Choose which minimizes the sum of MWPD to all other picking up time sequences in , denoted as,

$$\sum$$

Then, we got six classifiers , given a new time sequence for each classifier, if the MWPD to is no more than , then the corresponding classifier outputs 1, which means that belongs to the owner; otherwise outputs -1, which means that belongs to another person. See Table 1 for a summary of the DTW-based classifier learning.

However, in practice, a classifier obtained by using single axis of acceleration or angular speed is a weak learner, which is only slightly correlated with the true classification (in other words preforming slightly better than random guessing). A strong learner that is arbitrarily well correlated with the true classification is required.

D. Training Strong Classifier with Boosting

Boosting algorithm is motivated by the fact that it is easier to find a weak classifier with low accuracy rather than to find a strong classifier with high accuracy. The basic idea of Boosting is that weak classifiers can be integrated into a whole, to build a stronger classifier, so that the performance of the new classifier could be better than that of any of those weak ones. Suppose that are a set of weak learners, then the strong learner could be constructed as

Table 2: The AdaBoost algorithm for classifier learning

Algorithm 2: Two-class Discrete AdaBoost Algorithm

Input: given time sequences ( ) ( ) ( ) where , for negative and positive sequences respectively, and the j-th sample point in is

Initialization: Set the weight of m samples as

For

1. Normalize the weights,

   so that is a probability distribution.
2. For each dimension of time sequences, train a weak classifier which is restricted to using a single dimension . The error is evaluated with respect to
3. Choose the classifier with the lowest error
4. Calculate coefficient of classifiers:
5. Update sample distribution:

Output: the strong classifier: ( ) ( )

where represents the coefficient of and can be learned during the processing of Boosting.

Valiant et al. posed the question if a set of weak learners can be converted into a single strong learner [15]. Schapire proved the equivalence between weak learning and strong learning, and proposed a polynomial time Boosting algorithm [16]. Freund and Schapire provided an improvement of the original Boosting algorithm. Since no prior knowledge about the weak learners is needed, it is easier to be applied in practice [17][18].

In SmartDog, we adopt a Two-class Discrete AdaBoost Algorithm, the procedure is as follows: first, assign the same weight for all training samples. Suppose there are samples in the training set , the weight of each sample is initialized to . We call the weight assignment as sample distribution. Then start K rounds iteration. In each iteration, five steps are included. First, train several weak classifiers under the current sample distribution, and calculate the error rates of each classifier. Second, choose the classifier which holds the minimum error Third, Use to calculate the coefficient of the classifier. Forth, rebuild the sample distribution used in next round. Finally, normalize the weights. After the iteration, the final strong classifier is obtained as ( ). See Table 2 for a summary of the boosting process.

In practice, the inputs of Algorithm 2 are the picking up motion sequences from an owner and the stealers. If belongs to the owner, then , otherwise, Six weak classifiers built in Algorithm 1 are used to construct the strong classifier.

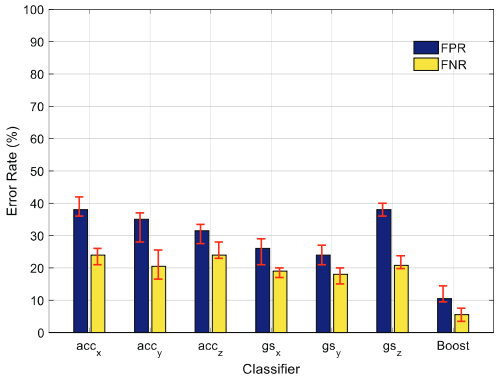

Figure 3. Performance of weak classifiers and the strong classifier obtained by conducting boosting.

VI. Evaluation

Using picking up samples collected from 20 volunteers, which has been described in Section III. According to the status of volunteers, picking up data can be divided into five scenarios: standing in shopping mall, standing on bus, metro and escalator, and sitting on chair. In every scenario, a smartphone is placed in jacket pocket, pants pocket and shoulder bag of a volunteer, respectively. Since there are 50 picking up signals available under each configuration, we randomly choose 35 of them to train the classifiers and 15 are used for verifying the performance of theft detection.

We evaluate the performance of SmartDog through trace driven simulations, considering two metrics, i.e., false positive rate (FPR), referring to the probability of treating a stealer as the legitimate user when testing, false negative rate (FNR), referring to the probability of rejecting the legitimate user when testing. For each experiment, we repeat that experiment for ten times and present the average error rates.

A. Overall Performance of SmartDog

In this experiment, we compare the overall performance of each weak classifiers and the strong classifier obtained by conducting boosting. In Figure 3, average FPR and FNR of all weak classifiers and the strong classifier are depicted by histograms with two bars representing maximum and minimum errors. The blue and yellow rectangles represent FPR and FNR, respectively. represents the weak classifier which is trained by using x-axis acceleration data. Boost represents the final strong classifier. It can be seen that the average FPR and FNR of weak classifiers are relatively high, which are 24% ~ 39% and 19% ~ 24%, respectively. Figure 3 also corroborates that by integrating weak classifiers into a whole, a more powerful classifier can be built, whose performance is much better than any of those weak ones. We also emphasize that the FNR is critical for theft detection, since such kind of errors result in phone lost. Our experiment results show that the FPR and FNR of the final strong classifier obtained by conducting

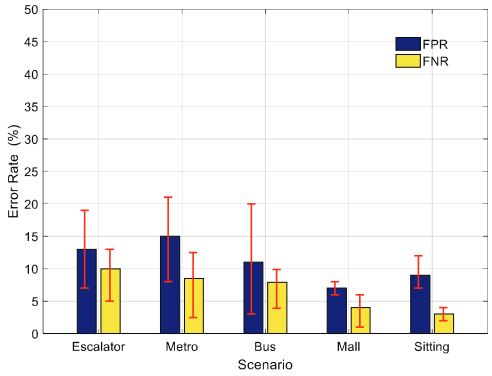

Figure 4. Average FPR and FNR under different scenarios.

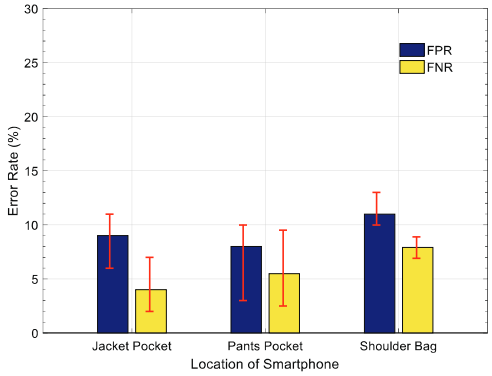

Figure 5. Average FPR and FNR when smarphones are placed in jacket, pants pockets and shoulder bags.

boosting are 10.2% and 5.5%, respectively. Thus SmartDog is of great help to smartphone protection.

B. *Impact of Volunteer Status and Locations of Smartphones*

We investigate the impact of volunteer status to the performance of SmartDog. We count the average FPR and FNR corresponding to five different statuses of volunteers, separately. The results are shown in Figure 4. It can be seen that when volunteers are standing in escalator, metro and bus, the FPR are relatively high, which are 13.5%, 14.9% and 11.2%. When volunteers are standing in shopping mall or sitting down somewhere, FPR and FNR are only approximate 6.8%, 9% and 4%, 3%. This implies that the mobility of transportations affects the accuracy of SmartDog slightly.

We also study the performance of SmartDog when smartphones are placed in different places, including jacket and pants pockets, and should bags. The results are shown in Figure 5. It can be seen that, in the three places, SmartDog has similar performance. The average FPR are approximate 9%, 7% and 11%, when smartphones are located in jackets, pants and bags, respectively, and the average FNR are approximate 4%, 5% and 8%. This verifies that SmartDog is very reliable to different places smartphones located.

VII. Conclusion

In this paper, we have proposed a real-time smartphone theft detection scheme, called SmartDog, based on the picking up motions of smartphone owners. SmartDog can actively protect the smartphone from being stolen, and it is resilient to shoulder-surfing and biometrics hacking attacks as it adopts both physiological and behavioral characteristics to profile owners. Furthermore, SmartDog is quite reliable and can work well with different modes of transport. As SmartDog needs only off-the-shelf devices, it is easy to gain a wide deployment. Nevertheless, SmartDog also has several limitations. For example, in order to achieve low false negative error rate, the false positive error rate is relatively high. Another limitation of SmartDog is that currently it can only work with two common people postures, i.e., sitting and standing. It would be more practical if more postures were supported.

VIII. Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61300199, 61672151, and 61402101); by the Fundamental Research Funds for the Central Universities (Grant No. 2232014D3-21, and 2232015D3-29); by Shanghai Municipal Natural Science Foundation (Grant No. 14ZR1400900).

References

[1] TrendForce, http://press.trendforce.com/press.html, Press Release.
[2] BBC news: 314 mobile phones 'stolen in London every day', http://www.bbc.com/news/uk-england-london-21018569.
[3] Norton Cybercrime Report 2011, http://us.norton.com/content/en/us/home_homeoffice/html/cybercrimereport/.
[4] Symantec, "2014 Symantec canada smartphone honey stick project", report, 2014.
[5] "Tencent Mobile Phone Manager," http://m.qq.com/anti_theft/login.jsp.
[6] "Rising Phone Security Assistant," http://mobile.rising.com.cn/android/.
[7] X. Yu, Z, Wang, L. Sun, W. Zhu, N. Gao, and J. Jing, "Remotely wiping sensitive data on stolen smartphones," in proceedings of the 9th ACM symposium on Information, computer and communications security, 2014, pp. 537-543.
[8] L. Subramanian, G Q. M. Jr., and P. Stephanow, "An architecture to provide cloud based security services for smartphones," in proceedings of 27th Meeting of the Wireless World Research Forum, 2011.
[9] A. U. S. Khan, M. N. Qureshi, and M. A. Qadeer, "Anti-theft application for android based devices," in proceedings of IEEE International Advance Computing Conference, 2014.
[10] Simon, R. Anderson, "Security analysis of consumer-grade anti-theft solutions provided by android mobile anti-virus apps," in proceedings of the 4th Mobile Security Technologies Workshop, 2015.
[11] "Software reduces identity theft risk in stolen cell phones," http://mobiledevdesign.com/news/software-reduces-identity-theft-risk-stolen-cell-phones.
[12] "Lookout Mobile Security," https://www.lookout.com.
[13] "360 Guard," http://shouji.360.cn.
[14] "Kingsoft Guard," http://www.ijinshan.com/hd/oneshow2012/osweb-sjws.htm.
[15] M. Kearns and L. G. Valiant. "Cryptographic limitation on learning Boolean formulate and finite automata," *Journal of the Association for Computing Machinery*,vol. 41, issue 1, pp. 67-95. January 1994.
[16] R. E. Schapire, "The strength of weak learn ability," *Machine Learning* vol. 5, issue 2, pp. 197-227. 1990.
[17] Y. Freund, "Boosting a weak learning algorithm by majority," *Information and Computation*, vol.121, issue 2, pp. 256-285, 1995.
[18] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, issue 1, pp. 119-139, August 1997.