Exploring Touch-Screen Biometrics for User Identification on Smart Phones

Julio Angulo, Erik Wästlund

▶ To cite this version:

Julio Angulo, Erik Wästlund. Exploring Touch-Screen Biometrics for User Identification on Smart Phones. 7th PrimeLife International Summer School (PRIMELIFE), Sep 2011, Trento, Italy. pp.130-143, 10.1007/978-3-642-31668-5_10 . hal-01517592

HAL Id: hal-01517592

https://inria.hal.science/hal-01517592

Submitted on 3 May 2017

Exploring Touch-screen Biometrics for User Identification on Smart Phones

Julio Angulo and Erik Wästlund

Karlstad University,
Universitetsgatan 2, 651 88 Karlstad, Sweden
{julio.angulo,erik.wastlund}@kau.se
http://www.kau.se

Abstract. The use of mobile smart devices for storing sensitive information and accessing online services is increasing. At the same time, methods for authenticating users into their devices and online services that are not only secure, but also privacy and user-friendly are needed. In this paper, we present our initial explorations of the use of *lock pattern* dynamics as a secure and user-friendly two-factor authentication method. We developed an application for the Android mobile platform to collect data on the way individuals draw lock patterns on a touch-screen. Using a Random Forest machine learning classifier this method achieves an average Equal Error Rate (EER) of approximately 10.39%, meaning that lock patterns biometrics can be used for identifying users towards their device, but could also pose a threat to privacy if the users' biometric information is handled outside their control.

Keywords: Mobile user experience, biometrics, smart mobile devices, mobile identity management, mobile authentication, privacy, lock patterns

1 Introduction

Smart mobile devices have become essential tools in many people's daily lives. Not only are we using these devices as the means to communicate with others, as sources of entertainment and as ways of expressing ourselves, but we also use them to store sensitive personal information and access different online services. Despite all the information contained in a device and the transactions that can be performed with it, many users choose not to protect their devices [6], and at the same time they tend to be perpetually logged into some of the services provided by mobile third party applications. Thus, an attack on the mobile device, or the loss of it, can have negative consequences, such as the intrusion of privacy, the opportunity to impersonate users, and even severe financial loss.

Currently, most of the solutions for authenticating users into their devices and other mobile services are based on the same solutions offered when using desktop computers, which usually involve the use of a PIN, a strong password, or some sort of extra external security token device. These techniques become cumbersome when applied to mobile devices and do not always provide a satisfactory user experience. Besides, they are not a sustainable approach for the future of mobile interactions, in which people would carry only one secure trustable device to perform most of their tasks and would preferably use only one hand to operate such device [17].

As a proposed solution to these issues we are investigating how screen *lock patterns* can be enhanced with the use of biometric features. By lock patterns we refer to the option contained in the Android mobile platform[1] for locking the phone's screen (Figure 1). Lock patterns are one type of recall-based graphical password [2]. They have been criticized because of their vulnerabilities to *smudge attacks* (i.e., recognizing the fingers' grease on the screen) [1], *shoulder-surfing attacks* [32] (i.e., observing or recording with a video camera the moment of authentication), and others. We hypothesize that adding biometric analysis to lock patterns can enhance the security of this type of graphical passwords by becoming a two-factor authentication mechanism. This method would also be privacy-friendly if used to protect the users' sensitive information stored locally, since the users' biometric features would be kept securely inside the device and under the users' control. However, if used for remote authentication, it could pose a threat to the anonymity of users.

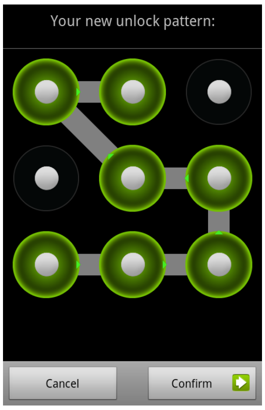


Fig. 1. An Android lock pattern

In this paper we first present in Section 2 work related to the study of graphical passwords and biometrics methods employed in modern mobile devices. Section 3 presents the identified requirements and research questions. Section 4 describes our experimental approach and collection of data. Section 6 describes the implications of our results and other reflections. Finally, Section 7 presents our plans for future investigations and conclusions.

2 Related work

Research has been done in exploring different biometric approaches for providing an extra level of security for authenticating users into their mobile devices. Specifically, research done on the analysis of keystroke dynamics for identifying users as they type on a mobile phone can be found in [7], [8], [16], [24], [33] and others. One of these studies, [7], considers the dynamics of typed 4-digit PIN codes, in which the researchers achieved an average Equal Error Rate (ERR)[2] of 8.5%. However, the data for this experiment was collected using a "mobile phone handset (Nokia 5110), interfaced to a PC through the keyboard connection " [7],

[1] http://developer.android.com/index.html

[2] The Equal Error Rate (EER) is a measurement used to compare different biometric systems, and is explained more in detail in Section 5

thus their experiment does not portray real mobile situations neither does it consider typing PIN codes on touch-screens. To the best of our knowledge, only one of the mentioned studies, [33], partially considers the use of on-screen keyboards. The approach taken in this study, however, has the disadvantage that the system has to be trained with a minimum of 250 keystrokes in order to achieve a low Equal Error Rate of approximately 2%, which is not suitable for applications that do not require a lot of typing, neither for detecting short passwords or PIN intrusions. From the literature, it is still uncertain that enhancing 4-digit PIN inputs with biometrics on touch-screens would provide higher levels of security for protecting sensitive information stored on mobile devices.

At the same time, imposing the use of alphanumeric passwords on mobile devices creates the problem that users tend to choose simpler, weaker or repetitive passwords [24], since complicated strong passwords are harder to type on smaller on-screen keyboards. Therefore, suggestions for more unobtrusive methods for authentication on mobile smart phones have emerged as an alternative to typed passwords, such as gait biometrics (achieving an EER of 20.1%) [10] [26], or the unique movement users perform when answering or placing a phone call (EER being between 4.5% and 9.5%) [9]. Although these methods seem to be a promising approach towards enhancing the user experience, they require users to take the explicit actions of walking or answering phone calls in order to be effective. Therefore, they are not fully suitable for scenarios when a user needs to interact or look at the phone in order to login to a mobile application or online service. Besides, these methods only provide a one-factor authentication mechanism.

Researchers have also suggested the use of graphical passwords as an easier alternative to written passwords, based on the idea that people have a better ability to recall images than texts. A good overview of popular graphical password schemes has been reported in [2]. Different usability studies have outlined the advantages of graphical passwords, such as their reasonable login and creation times, acceptable error rates, good general perception and reduced interference compared to text passwords, but also their vulnerabilities [27] [30]. As mentioned earlier, lock patterns are one type of recall-based graphical password [2]. To the best of our knowledge, the enhancement of lock patterns with biometric information has not been explored so far.

Regarding issues to users' anonymity, a recent study has demonstrated that pseudonyms chosen by users at different websites can be linked to deduce their real identity [28]. This probability of linking and profiling users could increase if users' pseudonyms can be identified based on their lock pattern behavior when authenticating into different services. Yet another study has shown that users' identities can be reconstructed from their typing patterns while browsing online [5]. In our paper, we consider if similar issues can arise when employing the biometrics of lock patterns for authentication.

3 Requirements and research questions

The initial motivation for our research arose from the need to provide secure and unobtrusive methods for authenticating users of mobile devices. We chose to explore the use of lock patterns for authentication since it has the benefits of not involving additional physical gadgets (such as secure external digital tokens) and not demanding the users' attention for a long period of time, but at the same time making users aware of their momentary intent. Also, lock patterns are less prone to repetitive errors compared to typing strong passwords on a touch-screen, and allow users to login while on the move in a more seamless manner. Besides the usability advantages, enhancing lock patterns with biometrics would improve this method's security by becoming a two-factor authentication method.

Although 4-digit PIN codes, which are currently the most common way of granting access to information stored on the device, provide similar benefits in terms of usability, it seems unlikely that their security could be enhanced with keystroke dynamics analyses on touch-screen devices, given that the size of a PIN code is too small and that previous studies have shown that large training sets are required to achieve good performances on touch-screens [33]. Furthermore, other studies have shown that it is more difficult for users to remember different PINs over extended periods of time than graphical passwords [23].

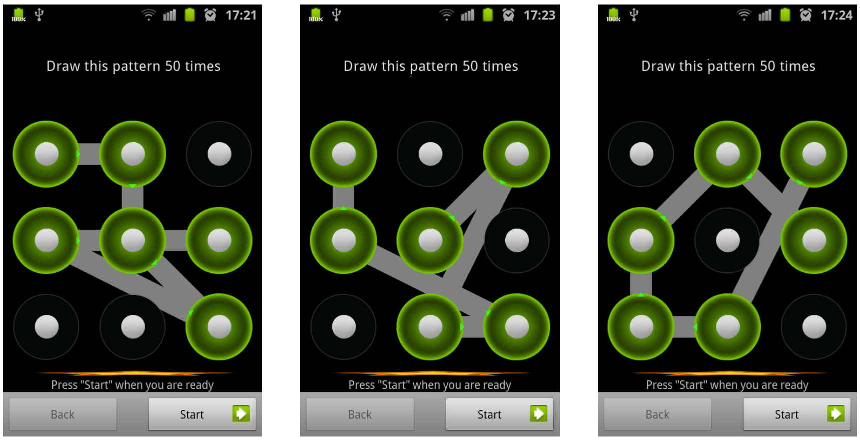
Our idea of using lock patterns biometrics for authentication raised the following research questions:

1. **Do lock patterns provide a set of distinguishing features that are unique to each individual?** Is it possible to verify the identity of individuals by the way they draw a pattern on the screen?
2. **What are the privacy challenges that need to be considered when using this authentication method?** If users can be uniquely identified by the biometrics of lock patterns, what are the privacy issues to be tackled before this method can be used in practice?

4 Experimental setup

Using Google's platform for mobile devices, Android [14], we have developed a mobile application to collect data from different individuals on the way they draw lock patterns, their experience while doing so and other contextual factors.

To find the answer to our first research question, test participants were asked to draw three different lock patterns correctly a certain number of times (n=50 trials for each pattern), with each pattern consisting of six dots, as shown in Figure 2. More specifically, during a test session test participants were first shown an animation on how to draw the first lock pattern (see Figure 2(a)), once they had learnt it they were asked to draw that pattern correctly 50 times. They were then shown the second pattern (Figure 2(b)) and were also asked to draw it 50 times, and the same was done for the third pattern (Figure 2(c)). A static approach was used in which all participants drew the same three patterns, i.e., the input was identical for all tests [7]. Analogous to earlier keystroke studies (in which different distinguishing features are used, such as key holding time and digraphs [33]), two main features were captured for each successful trial: the *finger-in-dot* time, which is the time in milliseconds from the moment the participant's finger touches a dot to the moment the finger is dragged outside the dot area, and the *finger-in-between-dots* time, representing the speed at which the finger moves from one dot to the next. All erroneous trials were disregarded.



Draw this pattern 50 times

Press "Start" when you are ready

Draw this pattern 50 times

Press "Start" when you are ready

Draw this pattern 50 times

Press "Start" when you are ready

(a) First lock pattern (b) Second lock pattern (c) Third lock pattern

Fig. 2. The three lock patterns that participants were asked to draw

5 Data collection and analysis

A total of 32 different participants completed the test successfully using the mobile application. This is comparable to the amount of test participants used in similar studies (e.g., [7], [9], [33], [34]). Participants were 12 women and 20 men, coming from different age groups (from 19 to 56 years old), cultural and educational backgrounds, and having different levels of experience interacting with touch-screen smart phones. The tests were performed with different Android phones: Samsung Galaxy SII (18), Nexus S (8), HTC Legend (4) and HTC Vision (2).

The data collected on the participants' finger movement times were used to calculate the common standard metrics used to assess biometric systems, the *False Acceptance Rate* (FAR) which indicates the probability that the system will erroneously grant access to an intruder, and the *False Rejection Rate* (FRR) which is the probability that the system will wrongly deny access to a legitimate user. The point at which both FAR and FFR are equal is denoted the *Equal Error Rate* (ERR). The EER makes it easier to compare the performance of various biometric systems or classifiers, and the lower its value the better the classifier. The Random Forest classifier, for instance, has been previously used to analyze the keystroke dynamics of users entering PIN codes on computer keyboards [22] [34]. Since the lock patterns presented here are composed of a 3x3 grid, which resembles the layout of a keypad, we trusted this algorithm to provide us with a good estimate for ERR. The Random Forest has the advantage of being useful for clustering and detecting outliers, as well as being robust against noise and having a fast learning process for large datasets. We used the implementation of the Random Forest algorithm from the R package v4.6-2 [3]. In order to compare performance, we present the results from other five classifiers previously used in keystroke analysis studies, also obtained with the R statistical program (v2.13.1). The Supportive Vector Machine (SVM) and Recursive Partitioning (RPart) classifiers are used in [34], whereas the Manhattan, the Nearest Neighbor (Mahalanobis) and Eucledian detectors are provided in [20] and [21][3]. We refer the reader to the corresponding publications for detailed explanations of these different algorithms.

From the collected data, we were left with six *finger-in-dot* variables and five *finger-in-between-dots* variables for each trial, making a total of eleven variables to feed the classifiers. As mentioned earlier, participants were asked to draw each of the three patterns, shown in Figures 2(a), 2(b) and 2(c), 50 times, leaving us with 150 trials in total. During our initial analysis of the data we decided to disregard the first 10 trials out of the 50, since we considered that each participant used those initial trials for their own practice (i.e., *human* learning trials). Then, with no further analysis of the data, we selected the next 25 trials for training the classifiers (i.e., *machine* learning or *training* trials) and the remaining 15 trials were used for testing (i.e., *testing* trials). Table 1 shows the obtained mean EER with their corresponding standard deviations for all six classifiers. As expected, the Random Forest classifier provided the best result, giving an average EER of 10.39% with a standard deviation of 3.0%.

Table 1. Obtained mean Equal Error Rates and standard deviations

	Eucledian	**Manhattan**	**Mahalanobis**	**RPart**	**SVM**	**RandomForest**
Mean EER	0.2734767	0.2559011	0.2302509	0.2968256	0.1406362	0.1039453
Standard deviation	0.098	0.094	0.097	0.096	0.057	0.03

More in-depth analysis of the data showed that there is a negative linear tendency between the number of training trials and the EER value obtained using the Random Forest classifier. Also, we observed that disregarding more of the initial trials (as human trials) up to a certain amount (25 trials) results in a better EER value (8.87%); indicating that the more comfortable or experienced users become when drawing a pattern, the better chances of correctly identifying them. Table 2 shows the EER values obtained using the Random Forest classifier when using different configurations for a varying number of trials, such as increasing the number of training trials while keeping the testing trials constant, or keeping the number of training trials constant while increasing the human trials and

[3] A script for the R platform is provided by Carnegie Mellon University, available at `http://www.cs.cmu.edu/~keystroke` (Accessed 2011-07-25)

decreasing the testing trials. The highlighted row on the table indicates the initial selected configuration, which is the configuration we have chosen to present in this paper. This configuration was selected since it represents a balanced number of trials that gives a relatively good EER estimate. Choosing a greater number of training trials would have an impact on usability, while decreasing it would result in a greater EER value.

Table 2. EER obtained by different configuration of the number of trials

			EER (RandomForest)	
Human trials	Training trials	Testing trials	Mean	Std dev
10	10	30	0.168590	0.062
10	15	25	0.148422	0.056
10	20	20	0.123632	0.037
10	25	15	0.103945	0.030
10	30	10	0.093223	0.032
10	35	5	0.079075	0.040
10	39	1	0.052462	0.069
10	10	10	0.137028	0.065
10	15	10	0.130543	0.065
10	20	10	0.115598	0.053
10	25	10	0.099589	0.033
10	30	10	0.092505	0.032
15	20	15	0.106840	0.033
20	20	10	0.096232	0.039
25	20	5	0.088665	0.043
5	20	5	0.165284	0.075
5	20	10	0.161635	0.062
5	20	15	0.155483	0.060
5	20	20	0.152889	0.056
5	20	25	0.145395	0.054
0	5	1	0.148098	0.113

Figure 3 shows the fitted Receiver Operating Characteristic (ROC) curve for all participants using the above mentioned configuration. This curve “allows the evaluation of different machine learning algorithms by measuring the rate of false positives and true positives against a varying threshold level” [25]. The ROC curve in this case provides us with the formula $y = x^{(0.051 \pm 0.002)}$, from where we can infer that having a FAR of 5% would give us a probability of correctly admitting a legitimate user (a True Acceptance Rate or TAR) between 85.32% and 86.35% ($y = 0.05^{(0.051 \pm 0.002)}$). Therefore the value for FRR ($FRR = 100\% - TAR$) lies between 13.65% and 14.68%.

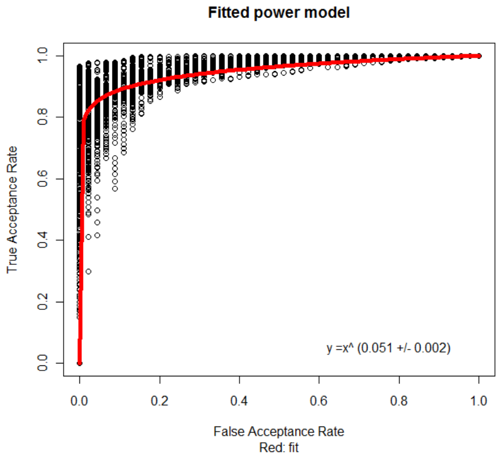


Fig. 3. Fitted ROC curve

Table 3 shows some other possible obtained values of FRR given predetermined values of FAR with the use of the ROC curve formula. This depicts the clear tradeoff between the usability (FRR) and the security (FAR) of the system.

Table 3. Calculated FFRs by given FARs using the formula $y = x^{(0.051 \pm 0.002)}$

FAR (x)	**TAR** (y)	**FRR** ($1.00 - y$)
0.05	[0.8532, 0.8635]	[0.1365, 0.1468]
0.10	[0.8851, 0.8933]	[0.1067, 0.1149]
0.15	[0.9043, 0.9112]	[0.0888, 0.0957]
0.20	[0.9182, 0.9242]	[0.0758, 0.0818]
0.25	[0.9292, 0.9343]	[0.0657, 0.0708]
0.30	[0.9382, 0.9427]	[0.0573, 0.0618]

Note that the values presented in Table 1, Table 2, and Table 3 are based on a binary classification of one subject drawing all of the three different patterns and for all 11 variables. However, analyzing each lock pattern separately the values shown in Table 4 are obtained. In order to investigate if there is any systematic difference between the Equal Error Rates obtained from three patterns, a repeated measure ANOVA was run. The results show that there is no significant main effect for the three patterns, $F(2, 62) = 0.021$ ($\rho = 0.979$). To exclude the possibility of this result being a Type II Error, a Power Analysis was performed with the use of R statistical software, which showed that, for a medium sized effect ($f = 0.25$) and sample of $n = 96$ (3 patterns for each 32 participants), the obtained power is .96 to find a significant effect at the $\sigma = .05$ level. In other words, the results show that individual error seems to be consistent over the tree patterns used in the study, implying that users can be identified regardless of the pattern they draw.

Table 4. Mean Equal Error Rates and standard deviations for each lock pattern

Lock pattern		**Eucledian**	**Manhattan**	**Mahalanobis**	**RPart**	**SVM**	**RandomForest**
First	Mean EER	0.220685	0.198577	0.22637	0.303662	0.131384	**0.099484**
	Std dev	0.111	0.106	0.136	0.134	0.091	0.061
Second	Mean EER	0.236102	0.210807	0.225255	0.343344	0.115054	**0.100343**
	Std dev	0.110	0.100	0.130	0.132	0.065	0.051
Third	Mean EER	0.199664	0.179404	0.185995	0.290088	0.125538	**0.102358**
	Std dev	0.094	0.085	0.105	0.128	0.076	0.062

6 Implications and discussions

The following sections present some of the security, privacy and usability implications of the results obtained by our experiments as well as other related discussions.

Improving the security of biometrically enhanced lock patterns. The results above suggest that lock pattern biometrics have the potential to be employed as an authentication method and that individuals can be identified by the way they draw a lock pattern on a touch-screen. We consider this result to be a good beginning on the exploration of touch-screen dynamics given that the moderately low EER (10.39%) was obtained without applying any other analytical enhancements to the data (such as handling outliers, differentiating distances between dots, optimizing the human learning effect, grouping data by device, etc.).

Applying dynamic optimizers to the data set could be a way to greatly improve the obtained EER value. For example, when looking at keystroke biometrics on mobile devices, the work presented in [33] initially reports an EER value greater than 26.5% using a RBFN (Radial Basis Function Neural network) classifier [15]. However, using fuzzy classifiers the researchers were able to lower the EER value to around 18.6%. Then they applied a hybrid version of the Particle Swarm Optimizer (PSO) [19] and Genetic Algorithm (GA) [13] to further lower their claimed EER value to less than 2.07%. We believe that applying similar optimizers to our data could similarly reduce our obtained EER value.

Moreover, the security of the lock pattern method is very dependent on the number of available pattern combinations. As it is currently implemented, the Android lock pattern mechanism consist of a 3x3 grid, not allowing repetition of dots and always including dots that lie in-between two other dots. However, since our data shows that users can be identified regardless of the pattern they draw on the screen, the existing lock pattern mechanism could be improved by removing the imposed constrains and considering a bigger grid, thus increasing the password space.

As part of our future work, we are also planning to explore if the security of such a system could be improved by measuring the *pressure* of the users' fingers on the touch-screen and the *tilting angle* in which a user holds a device, which can be used as additional biometric features to feed the machine learning classifiers.

Two- and three-factor authentication method. Note that the obtained EER (10.39%) was calculated on the assumption that an imposter already knows the user's secret pattern, thus working as a one-factor authentication. However, supposing that the pattern is only known to the legitimate user, the chances for an imposter to successfully authenticate into the system are further reduced. For example, given that there are 16,032 combinations of six-dotted patterns in the current implementation of the Android lock patterns, let the probability of an imposter entering the correct lock pattern on the first attempt to be $\Pr(PatternGuessing) = 1/16,032 = 0.00006$. Similarly, let the probability of the lock pattern biometric system to authenticate an imposter that knows the legitimate lock pattern, which is given by the value of the False Acceptance Rate (FAR), be $\Pr(FAR) = 0.05$. Thus, the probability of these two mutually independent events happening is given by

$$\Pr(PatternGuessing \cap FAR) = (0.00006) * (0.05) = 0.000003$$

In other words, this solution provides a two-factor authentication in which the probability that an attacker with an unknown pattern would be let into a system enhanced with biometrics is about 0.0003%, thus providing a much more secure solution than one-factor 4-digit and 5-digit PIN codes (0.01% and 0.001%). This probability can be further adjusted depending on the level of security required by a system.

What is more, this method could even provide three-factor authentication assuming that the mobile device has in place a Trusted Execution Environment (TEE) - or Mobile Trusted Module (MTM) - as described in [31] and considered in [12], [11] and [25]. Such TEE would basically guarantee that the reported state of a mobile device can be trusted by shielding dedicated pieces of its engine, thus providing one more level of security.

Therefore, the three security factors would be *something the user has* (a trusted mobile device), *something the user knows* (a secret lock pattern) and *something the user is* (the user's lock patterns biometrics). A three-factor authentication method could greatly reduce the chances of successful *smudge* attacks, *shoulder-surfing* and other common attacks.

Impact of training trials on performance. As with all biometric systems a number of training trials have to be input in order to accurately detect the identity of an individual. In the case of lock pattern biometrics, our analyses of the data presented in Table 2 shows that a set of 25 training trials tested against 15 testing trials provide a reasonably good EER value (10.39%). This implies that, if used in practice, a user would have to draw a pattern (or number of different patterns) 25 times on average for the system to achieve this level of efficiency, which has a possible impact on the usability of such system. However, this problem is solved by letting the user train the classifier with a reasonable amount of trials at the beginning (5 training trials gives an EER of 14.81%, as shown in the last row of Table 2) and use every subsequent authentication attempt that results in a successful login as an extra training trial. After having a robust set of training trials, an *aging factor*, as considered in [18], can also be introduced in which the oldest trials are weighted less and every new successful authentication attempt is taken as a new training trial. Thus the system would be adapting to the constant changes of the users and their environment.

Also, since our results show that users can be identified regardless of the way they drag their finger around the screen, the users' general interaction with the mobile device while dragging their finger can presumably be used to increase the original training set. However, this might have an impact on the processing power of the device.

Privacy friendliness. Now that we know that the security of lock patterns can be enhanced with biometrics, this mechanism can be employed for a number of security and privacy enhancing purposes, such as granting users access to some sensitive parts of a mobile application (e.g., a mobile banking application with different layers of security), authenticating users towards their locally stored sensitive private information, authorizing the use of encryption keys and (anonymous) credentials stored on the device, and so on.

In particular, this method could improve the way users give consent when, for example, engaging in online banking or mobile e-commerce transactions. Under these scenarios lock pattern biometrics could not only be employed to authenticate users towards their device, but also when the user is required to sign a transaction. Further explorations, such as the ones carried out as part of the U-PrIM project[4], are needed on incorporating the use of lock pattern biometrics with user-friendly mobile interfaces that make users aware of the actions they are taking depending on the context of a transaction.

Privacy unfriendliness. The findings presented here also raise some privacy concerns when interacting with touch-screen mobile devices, which bring us to our second research question, "*what are the privacy challenges that need to be considered when using this authentication method?*"

For one, third party applications could already be taking advantage of the fact that there is some degree of uniqueness in the way users drag their fingers across a touch-screen, thus being able to profile users and collect information as long as they are connected to a network and keep interacting with the application installed on their device. All of this happening in the application's background, without the users' awareness or consent.

Also, as mentioned previously, using this method to authenticate to remote online services could compromise the biometrics and privacy of users with a number of known attacks. Related to the work reported in [5], our results imply that, regardless of the users' choices of patterns for authenticating to different service providers, attackers would have a bigger probability to uncover the identity of pseudonymous users based on the biometrics of their secret lock patterns drawn at different websites, assuming that a website can get a hold of users' biometric data. Therefore, using lock patterns as a remote authentication method could result in linkability attacks and user profiling even when using different pseudonyms and patterns as passwords at different services' sites. Nevertheless, architectures have been proposed for authenticating smart phone users to remote web services in a privacy-friendly manner with the use of the previously mentioned Trusted Executing Environment (TEE) [24] [25]. This approach is also being considered within the U-PrIM project, where the existence of such TEE running on the mobile device is assumed, keeping the biometrics data secured under the users' control.

What is more alarming is the fact that common web browsers installed on mobile devices, such as Android's Browser, Apple's Safari, Firefox and others, allow the monitoring of users' swiping gestures directly on the web browser by running `JavaScript` code with the inherited touch and gestures events[5] (such as `touchstart`, `touchmove`, `touchend`, etc., as specified in [4] and [29]). Although these events allow users to browse the Internet in a more user-friendly manner

[4] U-PrIM (Usable Privacy-enhancing Identity Management for smart applications) `http://www.kau.se/en/computer-science/research/research-projects/u-prim`

[5] An example of the touch and drag events for Firefox on Android devices can be found at `http://www.quirksmode.org/m/tests/drag.html` (Accessed: 2011-10-27)

by dragging their finger around to pan a webpage or elements contained within a webpage, this implies that, by implementing these events, any common website could track the swiping movements of the users' fingers and collect the users' biometrics while they browse the Internet. However, this privacy threat can probably be diminished by using additional biometric features, such as the above mentioned *pressure* and *orientation* factors.

7 Conclusions and future work

The work presented here is our initial step towards finding user-friendlier methods for authentication into mobile smart devices. Our results show that adding biometric information to lock patterns can enhance the security of this method by providing two-factor authentication towards the smart device. A relatively low EER of 10.39% was achieved by analyzing the data from 32 individuals using a Random Forest classifier when combining the three different lock patterns and without any analytical enhancements to the data. This implies that users could be identified at this rate regardless of the pattern they draw. The levels of security and usability required by a system using lock pattern biometrics can be adjusted by varying the values of FFR and FAR over a threshold. Using this method, however, also raises some privacy issues that must be addressed before touch-screen dynamics can be used for authenticating remotely towards the services' sides. However, using a Trusted Executing Environment (also referred to as a Mobile Trusted Environment [31]) implemented in the device that protects users' biometric data, many of the privacy issues are reduced.

Our plans for future work include the application of other machine learning classifiers and analytical enhancements to get a better EER. Also, we would like to expand this study to include aditional biometric features, such as pressure and the tilting angle of the hand as the user draws a lock pattern. Work is also planned for exploring the contexts in which the mobile authentication takes place, since we believe that different contextual factors can have a great influence on the way lock patterns are drawn, and identifying these factors can improve the recognition of a legitimate user. Therefore, we plan to collect situational data that can give us an idea of the effect that context could have on the mobile authentication user experience.

Acknowledgments

This work is partly funded by the Swedish Knowledge Foundation (KK-stiftelsen) as part of the U-PrIM research project. We would like to acknowledge the kind help of Ge Zhang for providing us with the scripts for classifying our data. Also, our thanks go to Simone Fischer-Hübner and to the anonymous reviewers of the IFIP Summer School 2011 for their useful comments and advise.

References

1. Aviv, A.J., Gibson, K., Mossop, E., Blaze, M., Smith, J.M.: Smudge attacks on smartphone touch screens. In: Proceedings of the 4th USENIX conference on Offensive technologies. pp. 1–7. WOOT'10, USENIX Association, Berkeley, CA, USA (2010)
2. Biddle, R., Chiasson, S., van Oorschot, P.: Graphical passwords: Learning from the first twelve years. Technical report TR-11-01, School of Computer Science, Carleton University (January 2011)
3. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
4. Brubeck, M., Schepers, D., Moon, S.: Touch events version 1 - w3c working draft 13 september 2011 (September 2011), `http://www.w3.org/TR/2011/WD-touch-events-20110913/`, accessed 2011-10-27
5. Chairunnanda, P., Pham, N., Hengartner, U.: Privacy: Gone with the Typing! Identifying Web Users by Their Typing Pattern. In: 4th Hot Topics in Privacy Enhancing Technologies (HotPETs). The 11th Privacy Enhancing Technologies Symposium, Springer, Waterloo, Canada (July 2011)
6. Clarke, N.L., Furnell, S.: Authentication of users on mobile telephones - a survey of attitudes and practices. Computers & Security 24(7), 519–527 (2005)
7. Clarke, N.L., Furnell, S.: Authenticating mobile phone users using keystroke analysis. Int. J. Inf. Sec. 6(1), 1–14 (2007)
8. Clarke, N.L., Karatzouni, S., Furnell, S.: Flexible and transparent user authentication for mobile devices. In: SEC. pp. 1–12 (2009)
9. Conti, M., Zachia-Zlatea, I., Crispo, B.: Mind how you answer me!: transparently authenticating the user of a smartphone when answering or placing a call. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. pp. 249–259. ASIACCS '11, ACM, New York, NY, USA (2011)
10. Derawi, M.O., Nickel, C., Bours, P., Busch, C.: Unobtrusive user-authentication on mobile phones using biometric gait recognition. In: Proceedings of the 2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing. pp. 306–311. IIH-MSP '10, IEEE Computer Society, USA (2010)
11. Ekberg, J.E.: Mobile trusted computing based on MTM. IJDTIS 1(4), 25–42 (2010)
12. Ekberg, J.E., Bugiel, S.: Trust in a small package: minimized MRTM software implementation for mobile secure environments. In: STC. pp. 9–18 (2009)
13. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Boston, MA, USA, 1st edn. (1989)
14. Google: Android: Android - open source project (June 2011), `http://source.android.com/`
15. Hwang, Y.S., Bang, S.Y.: An efficient method to construct a radial basis function neural network classifier. Neural Netw. 10, 1495–1503 (November 1997)
16. Karatzouni, S., Clarke, N.L.: Keystroke analysis for thumb-based keyboards on mobile devices. In: SEC. pp. 253–263 (2007)
17. Karlson, A.K., Bederson, B.B., Contreras-Vidal, J.L.: Understanding Single-Handed Mobile Device Interaction (2006)
18. Kekre, H., Bharadi, V.: Ageing adaptation for multimodal biometrics using adaptive feature set update algorithm. IEEE International Advance Computing Conference pp. 535–540 (2009)
19. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks. pp. 1942–1948 (1995)

20. Killourhy, K., Maxion, R.: Why did my detector do that?!: Predicting keystroke-dynamics error rates. In: Proceedings of the 13th international conference on Recent advances in intrusion detection. pp. 256–276. RAID'10, Springer-Verlag, Berlin, Heidelberg (2010)
21. Killourhy, K.S., Maxion, R.A.: Comparing anomaly-detection algorithms for keystroke dynamics. In: IEEE Computer Society Press, Los Alamitos, C. (ed.) Proceedings of the 2009 IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2009. pp. 125–134. Lisbon, Portugal (June 2009)
22. Maxion, R.A., Killourhy, K.S.: Keystroke biometrics with number-pad input. Dependable Systems and Networks, International Conference on 0, 201–210 (2010)
23. Moncur, W., Leplâtre, G.: Pictures at the ATM: exploring the usability of multiple graphical passwords. In: Proceedings of the SIGCHI conference on Human factors in computing systems. pp. 887–894. CHI '07, ACM, New York, NY, USA (2007)
24. Nauman, M., Ali, T.: TOKEN: Trustable Keystroke-Based Authentication for Web-Based Applications on Smartphones. In: Bandyopadhyay, S.K., Adi, W., Kim, T.h., Xiao, Y. (eds.) Information Security and Assurance, Communications in Computer and Information Science, vol. 76, pp. 286–297. Springer Berlin (2010)
25. Nauman, M., Ali, T., Rauf, A.: Using trusted computing for privacy preserving keystroke-based authentication in smartphones. Telecommunication Systems pp. 1–13 (2011)
26. Nickel, C., Derawi, M.O., Bours, P., Busch, C.: Scenario test of accelerometer-based biometric gait recognition. In: Security and Communication Networks (IWSCN). 3rd International Workshop, Gjøvik, Norway (2011)
27. van Oorschot, P.C., Salehi-Abari, A., Thorpe, J.: Purely automated attacks on passpoints-style graphical passwords. IEEE Transactions on Information Forensics and Security 5, 393–405 (2010)
28. Perito, D., Castelluccia, C., Kâafar, M.A., Manils, P.: How unique and traceable are usernames? CoRR abs/1101.5578 (2011)
29. Safary Developer Library: Handling events (2011), `http://developer.apple.com/library/safari/#documentation/appleapplications/reference/SafariWebContent/HandlingEvents/HandlingEvents.html#//apple_ref/doc/uid/TP40006511-SW1`, accessed 2011-10-27
30. Salehi-Abari, A., Thorpe, J., van Oorschot, P.: On purely automated attacks and click-based graphical passwords. Computer Security Applications Conference, Annual 0, 111–120 (2008)
31. Trusted Computing Group: Mobile trusted module 2.0 - Use cases (March 2011), `http://www.trustedcomputinggroup.org/resources/mobile_trusted_module_20_use_cases`
32. Wiedenbeck, S., Waters, J., Sobrado, L., Birget, J.C.: Design and evaluation of a shoulder-surfing resistant graphical password scheme. In: Proceedings of the working conference on Advanced visual interfaces. pp. 177–184. AVI '06, ACM, New York, NY, USA (2006)
33. Zahid, S., Shahzad, M., Khayam, S., Farooq, M.: Keystroke-based user identification on smart phones. In: Kirda, E., Jha, S., Balzarotti, D. (eds.) Recent Advances in Intrusion Detection, Lecture Notes in Computer Science, vol. 5758, pp. 224–243. Springer Berlin / Heidelberg (2009)
34. Zhang, G.: Analyzing Key-Click Patterns of PIN Input for Recognizing VoIP Users. In: Camenisch, J., Fischer-Hübner, S., Murayama, Y., Portmann, A., Rieder, C. (eds.) Future Challenges in Security and Privacy for Academia and Industry, IFIP Advances in Information and Communication Technology, vol. 354, pp. 247–258. Springer Boston (2011)