Identification of Pressed Keys from Mechanical Vibrations

Gerson de Souza Faria, Hae Yong Kim

Abstract

This paper describes an attack that identifies the sequence of keystrokes analyzing mechanical vibrations generated by the act of pressing keys. We use accelerometers as vibration sensors. The apparatus necessary for this attack is inexpensive and can be unobtrusively embedded within the target equipment. We tested the proposed attack on an ATM keypad and a PIN-pad. We achieved the key recognition rates of 98.4% in ATM keypad, 76.7% in PIN-pad resting on a hard surface and 82.1% in PIN-pad hold in hand.

Index Terms

accelerometer, ATM, information security, side-channel attack, vibration, PIN-pad, point of sale.

I. Introduction

Nowadays, mechanical keyboards are the most common human-machine interface used in public places, due to their easy operation, robustness, simplicity and low cost. Mechanical keypads are the natural choice for entering confidential data in electronic payments, such as passwords in ATM (automatic teller machine) or in PIN-pads (a device used in smart card transactions to input the cardholder's personal identification number). In some countries, including Brazil, electors use electronic voting machines with mechanical keyboards to choose the candidate. Thus, the possibility that someone finds out the sequence of pressed keys, without the owner's knowledge or consent, is a serious security threat.

The authors are with the Departamento de Engenharia de Sistemas Eletrônicos, Escola Politécnica, Universidade de São Paulo, Av. Prof. Luciano Gualberto, tr. 3, 158, CEP 05508-010, São Paulo, Brazil (e-mails: gerson.faria@usp.br, hae@lps.usp.br)

In this paper, we show that devices based on the act of mechanically pressing keys have an intrinsic vulnerability. The act of pressing a key emanates mechanical energy as vibration that may be used to identify the pressed key. We tested our attack on two targets: an ATM keypad and a PIN-pad (also known as “POS - point of sale terminal” or “chip and PIN terminal”). The PIN-pad was tested in two situations: resting on a rigid surface and held on hand. The vibrations are acquired by accelerometers placed near the ATM keypad or inside the PIN-pad. All the experiments showed the same vulnerability.

Usually, modern ATM keypads are encrypted. They are sealed modules that encrypt the PIN soon after the entry. So, non-encrypted PIN numbers are not meant to be accessible from outside either by physically tapping onto wires or remotely sensing electromagnetic radiation. Any tampering of the keypad causes it to permanently disable itself. Similarly, PIN-pads are protected modules that permanently disable themselves if tampered. The possibility of identifying the PIN number through mechanical vibrations is a serious security failure to ATM keypads and PIN-pads because they are designed to resist against any attempt of eavesdropping. The devices will continue functioning normally while passwords are stolen. In our experiments, we used a protected PIN-pad but a non-sealed ATM keypad. However, we think that sealed ATM keypads leak the same kind of vibration.

The attack we present in this paper is considered a side channel attack. In cryptography, a side channel attack is any attack based on information gained from the physical implementation of a cryptosystem [1], [2]. For example, timing information, power consumption, electromagnetic or sound leak can provide extra information that can be exploited to break the system. Kuhn [3] considered the possibility of identifying the pressed keys by analyzing the resulting forces in some points of the equipment, e.g. by installing pressure sensors underneath the feet of the keyboard. He states that information theft through side-channels is unlikely to remain restricted to the classic electromagnetic, optical and acoustic domains. We do not exploit the forces measured under the terminal’s feet, but the vibrations generated when the keys are pressed.

The rest of the paper is organized as follows. The related works are described in Section II. The adopted approaches regarding signal analysis, feature extraction and classification are described in Section III. The attack to the ATM keypad is described in Section IV and the two experiments with the PIN-pad in Sections V and VI. Finally, we present our conclusions in Section VII.

II. Related Works

In the literature, there are some attacks that use information leaked in acoustic form. Some of them try to steal information by analyzing the sound generated by keystrokes of computer keyboards [4], [5],

[6], [7]; by keystrokes of ATM keypads [7]; and also by other devices such as dot matrix printers [8]. Halevi and Saxena describe eavesdropping attacks on pairing process of wireless devices that use acoustic “out-of-band” auxiliary channels [9]. Shamir and Tromer present a cryptanalysis proof-of-concept based on sound emanations from personal computers [10].

There are also some attacks that use the inner motion sensors of smartphones and tablets. Cai and Chen expound an attack that infers the pressed digits in an Android smartphone by analyzing the data from the equipment’s accelerometer at the moment the panel is touched [11]. Miluzzo *et al.* present a similar attack that uses accelerometer and gyroscope readings of smartphones and tablets [12]. A broad analysis on smartphone “sensor-sniffing attacks” is presented in [13].

On the other hand, there are very few previous works that try to identify the keystrokes through vibration analysis. Seemingly, the main consortium responsible for developing the standards of payment card security does not consider mechanical vibrations explicitly as a potential source of security threat.[1] Indeed, we published a preliminary version of our paper in a local conference [15] without being aware of any previous work. We became aware of the existence of another paper on the subject, written by Marquardt *et al.* [16] only after fully developing our own work. Marquardt *et al.* put a mobile phone near the target computer keyboard, both devices placed on a wooden desk. The phone’s accelerometer detects the vibrations generated by the keystrokes. These vibrations are used to classify keystrokes in “left” and “right.” Left keys are those located at the left side of the keyboard, including keys ‘t’, ‘g’ and ‘b’; and right keys are those located at the right side, including keys ‘y’, ‘h’, and ‘n’. They also classify the spatial distance between two consecutive keystrokes as “near” and “far.” For example, the two consecutive keystrokes “ca” is considered “near,” whereas the consecutive keystrokes “oe” is considered “far.” They achieved a correct identification of 84%-91% for left/right classification and 65%-70% for near/far classification.

We list below some differences between the Marquardt *et al.*’s work and ours. (1) Marquardt *et al.* do not properly identify the pressed key, but classify the keystrokes in left/right and near/far. They guess the typed English words with the help of an English dictionary. Our work identifies the pressed key in a numeric ATM keypad and a PIN-pad directly. (2) Marquardt *et al.* use only one accelerometer available inside iPhone, with low sampling rate. We use two or three accelerometers with higher sampling rates.

[1] “There is no feasible way to determine any entered and internally transmitted PIN digit by monitoring sound, electromagnetic emissions, power consumption or any other external characteristic available for monitoring - even with the cooperation of the device operator or sales clerk - without requiring an attack potential of at least 26 for identification and initial exploitation with a minimum of 13 for exploitation [14].”

As consequence, we achieve higher accuracy: in ATM experiment, we correctly identify the pressed key (choosing one out of nine possibilities) with recognition rate of 98.4%. Marquardt *et al.* achieve at most 91% in classifying the keystroke in left/right (choosing one out of two possibilities). (3) Marquardt *et al.* use a set of time and frequency domain features to characterize the keystrokes, including root mean square, skewness, variance, kurtosis, Fourier transform and mel-frequency cepstrum coefficients. We use only the correlations between time-shifted vectors of acceleration readings as features, because we noted that vibration responses can vary greatly, even when the same key is pressed by the same person (Fig. 1).

III. Adopted Approaches

The same person pressing the same key in his usual manner can produce very distinct vibration responses. We did some tests where a person pressed many times the same key in his normal way. Although he did not notice any major change in his pressing behavior, the resulting acceleration peaks ranged from 0.5g to 2g. Fig. 1 shows one such case, where the measured acceleration peaks greatly differ. Moreover, the two graphs do not differ only in amplitudes, but also in shapes: graph (b) has much more high frequency components than graph (a). We noticed that high frequency oscillations may appear and disappear in the vibration response of the same key. We hypothesize that this apparently odd phenomenon is due to the mechanical complexity of keyboards. If some acceleration threshold is surpassed, device-dependent nonlinearities may trigger spurious oscillations. As consequence, we do not use representations that characterize a signal (such as frequency spectrum, cepstrum, autoregression coefficients, wavelet coefficients etc.) as features. Instead, we use the features that describe mutual dependencies between two time-shifted versions of signals. The mutual dependencies are computed using zero-normalized cross correlations (NCC). Actually, we have tested many other features before concluding that NCC is the most appropriate one [15].

A. Notations

Let the column vector v with elements v_i, $0 \leq i < N$ represent the time series of N acceleration values captured by an axis of an accelerometer, with a fixed sampling period T. The mean value of v is denoted as $\bar{v}$. The mean-corrected vector $\tilde{v}$ has elements $\tilde{v}_i = v_i - \bar{v}$. We use only mean-corrected acceleration values, because we are not interested in the contribution of the static acceleration of gravity.

The dot product between two mean-corrected vectors $\tilde{v}$ and $\tilde{w}$ is:

$$\tilde{v} \cdot \tilde{w} = \sum_{i=0}^{N-1} \tilde{v}_i \tilde{w}_i \tag{1}$$

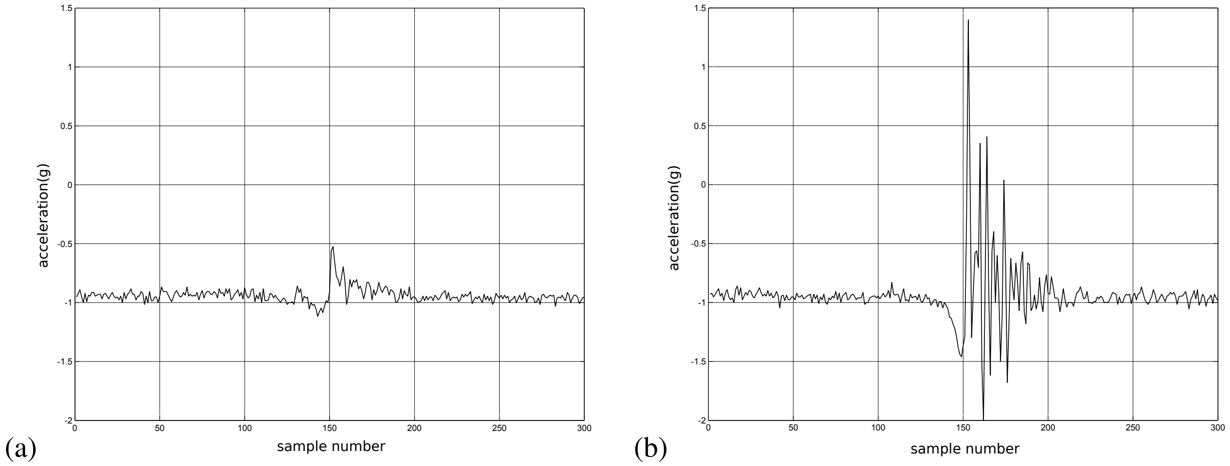

Fig. 1. Two keystrokes of the same key by the same person can be quite different. Both graphs represent the $\vec{z}$ axis acceleration values of A1 sensor in PIN1 experiment.

Dot product is directly related to the cosine of the angle between the two vectors and can be used to measure the "similarity" of the two signals. It is invariant to bias (because the vectors are mean-corrected) but it is not invariant to gain (it is directly proportional to the amplitudes of the two vectors). Dot product becomes invariant to amplitude (besides being invariant to bias) if we divide it by the norms of the two vectors, obtaining the correlation coefficient:

$$corr(\tilde{v}, \tilde{w}) = \frac{\tilde{v} \cdot \tilde{w}}{\|\tilde{v}\| \|\tilde{w}\|} \tag{2}$$

In an oversimplified scenario, suppose that the vibration generated by a keystroke takes n_1 sampling periods to reach the accelerometer A_1 and takes n_2 sampling periods to reach the accelerometer A_2. In this case, we will observe a peak in correlation between the acceleration values obtained by A_1 and those obtained by A_2, when the latter is shifted right $n_1 - n_2$ positions, thus characterizing the pressed key. In a real scenario, the behavior of vibrations generated by a keystroke is much more complex than this oversimplified reasoning. Even so, our experiments show that time-shifted correlation coefficients are very appropriate features to identify the pressed key.

Zero-normalized cross correlation (NCC) between vectors v and w (each with N elements) is a vector denoted as $\tilde{v} \otimes \tilde{w}$ whose elements are the correlation coefficients computed between time-shifted vectors, ignoring the elements that do not have the matching pair. It has $2N - 1$ elements:

$$(\tilde{v} \otimes \tilde{w})_n = \begin{cases} \dfrac{\sum\limits_{i=0}^{N-n-1} \tilde{v}_i \tilde{w}_{n+i}}{\sqrt{\sum\limits_{i=0}^{N-n-1} \tilde{v}_i^2 \sum\limits_{i=0}^{N-n-1} \tilde{w}_{n+i}^2}}, & 0 \leq n < N \\ (\tilde{w} \otimes \tilde{v})_{-n}, & -N < n < 0 \end{cases} \tag{3}$$

Note that $(\tilde{v} \otimes \tilde{w})_0 = corr(\tilde{v}, \tilde{w})$. NCC has been used for a long time in computer vision to find templates in search images, in an operation called template matching [17]. NCC has too many elements and it is not feasible to feed a machine learning algorithm with them all. So, we feed the learning algorithms with only a certain number of significant elements of NCC.

Let us define the (M, s)-correlation $[\tilde{v} \otimes \tilde{w}]_s^M$ between v and w as the column vector with $2M+1$ central elements of $\tilde{v} \otimes \tilde{w}$ in s step, for some $M \geq 0$ and $s \geq 1$:

$$[\tilde{v} \otimes \tilde{w}]_s^M = \begin{bmatrix} (\tilde{v} \otimes \tilde{w})_{-Ms} \\ (\tilde{v} \otimes \tilde{w})_{(-M+1)s} \\ \dots \\ (\tilde{v} \otimes \tilde{w})_{(M-1)s} \\ (\tilde{v} \otimes \tilde{w})_{Ms} \end{bmatrix}. \tag{4}$$

In other words, $[\tilde{v} \otimes \tilde{w}]_s^M$ is a subset of $2M+1$ central elements of $\tilde{v} \otimes \tilde{w}$ so that two neighboring elements in $[\tilde{v} \otimes \tilde{w}]_s^M$ are separated by distance s in $\tilde{v} \otimes \tilde{w}$. For example, for $M = 2$ and $s = 1$:

$$[\tilde{v} \otimes \tilde{w}]_1^2 = \begin{bmatrix} (\tilde{v} \otimes \tilde{w})_{-2} \\ (\tilde{v} \otimes \tilde{w})_{-1} \\ (\tilde{v} \otimes \tilde{w})_0 \\ (\tilde{v} \otimes \tilde{w})_1 \\ (\tilde{v} \otimes \tilde{w})_2 \end{bmatrix}. \tag{5}$$

and for $M = 2$ and $s = 2$:

$$[\tilde{v} \otimes \tilde{w}]_2^2 = \begin{bmatrix} (\tilde{v} \otimes \tilde{w})_{-4} \\ (\tilde{v} \otimes \tilde{w})_{-2} \\ (\tilde{v} \otimes \tilde{w})_0 \\ (\tilde{v} \otimes \tilde{w})_2 \\ (\tilde{v} \otimes \tilde{w})_4 \end{bmatrix}. \tag{6}$$

We will use this subset of $\tilde{v} \otimes \tilde{w}$ as features.

In ATM experiment, we use three analog accelerometers and take only the samples from $\vec{z}$ axis of each sensor, obtaining three signals. Thus, a keystroke is represented by three column vectors, each with 300 acceleration values:

$$\mathbf{V_{ATM}} = (\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3) \tag{7}$$

We chose to use only $\vec{z}$ axis because: (i) we had to minimize the amount of data because we did not use buffers and transmitted the samples in real time to the host computer; (ii) we conjectured that the most relevant component may lie at the same direction of the act of pressing the key.

In PIN-pad experiments, we use two digital accelerometers and take the samples from the three axes of each sensor. Thus, a keystroke is represented by six column vectors, each with 300 samples:

$$\mathbf{V_{PIN}} = (\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1, \mathbf{x}_2, \mathbf{y}_2, \mathbf{z}_2) \tag{8}$$

where $\mathbf{x}_1$ corresponds to the acceleration values from $\vec{x}$ axis of sensor 1 and so on. As we will see in sections V and VI, digital accelerometers have many characteristics that are not appropriate for our application. However, we had no way of knowing this fact when we designed the experiments.

B. Features

We use superscripts "ATM" or "PIN" to indicate that the feature can be applied to that specific experiment. We use superscript • to indicate that the feature can be applied to both experiments.

1) $\mathbf{same}^{\bullet}$ *:* Feature vector $\mathbf{same}^{\bullet}$ is the set of (M, s)-correlations of 2-combinations of acceleration vectors with the same physical axes. Writing it explicitly for both experiments:

$$\mathbf{same}^{\mathbf{ATM}} = \begin{bmatrix} [\tilde{\mathbf{z}}_1 \otimes \tilde{\mathbf{z}}_2]_s^M \\ [\tilde{\mathbf{z}}_2 \otimes \tilde{\mathbf{z}}_3]_s^M \\ [\tilde{\mathbf{z}}_1 \otimes \tilde{\mathbf{z}}_3]_s^M \end{bmatrix}, \ \mathbf{same}^{\mathbf{PIN}} = \begin{bmatrix} [\tilde{\mathbf{x}}_1 \otimes \tilde{\mathbf{x}}_2]_s^M \\ [\tilde{\mathbf{y}}_1 \otimes \tilde{\mathbf{y}}_2]_s^M \\ [\tilde{\mathbf{z}}_1 \otimes \tilde{\mathbf{z}}_2]_s^M \end{bmatrix} \tag{9}$$

This feature vector has a total dimension of $3 \times (2M + 1)$ elements. The rationale of using the NCC of vectors of the same physical axes is that different accelerometers will sense similar vibrations in the same axis with different delays. These delays may help to characterize the pressed key.

2) $\mathbf{comb^{PIN}}$ *:* Feature vector $\mathbf{comb^{PIN}}$ is the set of (M, s)-correlations of all 2-combinations of acceleration vectors. Here, we use the correlations even between different axes, hoping that the machine learning algorithm will correctly identify the pressed key when more information is available. It has a total dimension of $15 \times (2M + 1)$ elements, that is:

$$\mathbf{comb^{PIN}} = \begin{bmatrix} [\tilde{\mathbf{x}}_1 \otimes \tilde{\mathbf{y}}_1]_s^M \\ [\tilde{\mathbf{x}}_1 \otimes \tilde{\mathbf{z}}_1]_s^M \\ [\tilde{\mathbf{x}}_1 \otimes \tilde{\mathbf{x}}_2]_s^M \\ [\tilde{\mathbf{x}}_1 \otimes \tilde{\mathbf{y}}_2]_s^M \\ [\tilde{\mathbf{x}}_1 \otimes \tilde{\mathbf{z}}_2]_s^M \\ [\tilde{\mathbf{y}}_1 \otimes \tilde{\mathbf{z}}_1]_s^M \\ [\tilde{\mathbf{y}}_1 \otimes \tilde{\mathbf{x}}_2]_s^M \\ [\tilde{\mathbf{y}}_1 \otimes \tilde{\mathbf{y}}_2]_s^M \\ [\tilde{\mathbf{y}}_1 \otimes \tilde{\mathbf{z}}_2]_s^M \\ [\tilde{\mathbf{z}}_1 \otimes \tilde{\mathbf{x}}_2]_s^M \\ [\tilde{\mathbf{z}}_1 \otimes \tilde{\mathbf{y}}_2]_s^M \\ [\tilde{\mathbf{z}}_1 \otimes \tilde{\mathbf{z}}_2]_s^M \\ [\tilde{\mathbf{x}}_2 \otimes \tilde{\mathbf{y}}_2]_s^M \\ [\tilde{\mathbf{x}}_2 \otimes \tilde{\mathbf{z}}_2]_s^M \\ [\tilde{\mathbf{y}}_2 \otimes \tilde{\mathbf{z}}_2]_s^M \end{bmatrix}. \tag{10}$$

This feature is not used in ATM experiment because it uses only one physical axis ($\vec{z}$) and consequently $\mathbf{comb^{ATM}} = \mathbf{same^{ATM}}$.

C. Classification Schemes

In all experiments, the same data were used in three different “schemes.” The first scheme consisted on finding the row of the key, the second on finding its column, and the third on identifying the key directly. For instance, key “6” received label “2” in row scheme (because it is located in the second row), “3” in column scheme (it is located in the third column) and “6” in the key scheme. In ATM experiment, 9 keys distributed in 3 rows and 3 columns were used (Fig. 2 and Fig. 3) and in PIN experiments, 12 keys distributed in 4 rows and 3 columns were used (Fig. 4).

D. Machine Learning Algorithms

We used two machine learning algorithms: Multilayer Perceptron (MLP) and Support Vector Machine (SVM). In both cases, we used the implementations provided by OpenCV library [18]. No feature

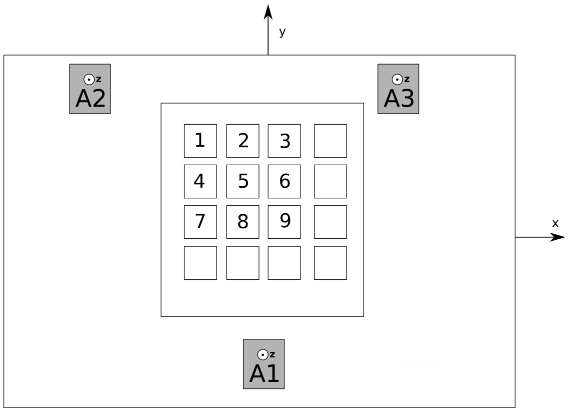

Fig. 2. Diagram depicting the ATM keypad experiment assembly. All the parts are fixed on a 4mm thick acrylic plate (the outer rectangle). Boxes labeled A1, A2 and A3 indicate the approximate positions of the three accelerometers.

dimension reduction technique was used. We also have tested some other machine learning algorithms before concluding that MLP and SVM are the most appropriate ones [15].

1) Multilayer Perceptron: MLP [19] configuration was composed of an input layer (the features themselves), two hidden layers with 30 neurons each and an output layer. The sizes of the output layers were 3 in column scheme; 3 (ATM) or 4 (PIN) in row scheme; and 9 (ATM) or 12 (PIN) in key scheme. Error back propagation was used as the learning algorithm. No network architecture optimization was executed.

2) Support Vector Machine: We used SVM with radial basis functions [20]. We let the OpenCV implementation choose automatically the optimal parameters.

IV. ATM Experiment

A. Experiment Assembly

For this attack, we fixed the keypad and the three accelerometers on a 4mm thick acrylic plate using screws, as depicted in Fig. 2.

We used a non-encrypted keypad with plastic keys, without any characteristic that would make it especially vulnerable (Fig. 3). This kind of keypad was widely used in self-service banking terminals some years ago. In some cases, this model has been replaced by encrypted keypads with metallic keys. However, we conjecture that metallic keys also leak similar mechanical vibrations.

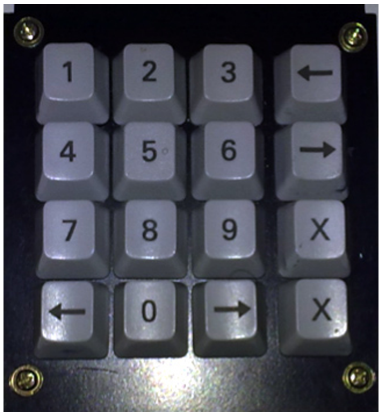

Fig. 3. A common ATM keypad was used in the attack.

Four small rubber feet were used to support the acrylic plate. We used Freescale MMA7260QT analog triaxial low-g accelerometers [21]. We used cables to connect them to the acquisition system, an ARM7 development toolkit based on a LPC2148 microcontroller. We used acceleration range of ±1.5g and sampling rate of 6700 samples/s with 10 bits resolution. For each keystroke, we acquired 300 samples from the $\vec{z}$ axis of each sensor, totalizing 3×300 samples. As we said above, we chose to use only the $\vec{z}$ axis because: (i) we conjectured that the most relevant component may lie at the same direction of the act of pressing the key and (ii) we had to minimize the amount of data because we did not use buffers and transmitted the samples in real time to the host computer. Further data processing was carried out on a PC.

B. Data Acquisition

In order to capture different ways of pressing the keys, each of 5 people pressed 10 times each of the 9 keys, resulting in 450 keystrokes. We discarded 9 of them due to acquisition errors, yielding 441 valid keystrokes (49 for each key). We divided them randomly in 7 groups, where each group contains 63 keystrokes, 7 for each key. The data from 6 out of 7 groups were used for the training. We reserved the data from the remaining group for the testing. This type of validation method is known as "k-fold cross-validation" or "rotation estimation" [24]. When $k = 1$, as in our case, it is also known as "leave-one-out" cross-validation method.

TABLE I

Recognition rate in % obtained in ATM experiment. Step s = 2

		Row Scheme		Column Scheme		Key Scheme	
Feature	Dim	MLP	SVM	MLP	SVM	MLP	SVM
same	27	99.8	99.8	97.4	97.8	96.4	97.8
	45	$\boxed{100.0}$	$\boxed{100.0}$	97.6	97.8	96.9	98.2
	63	$\boxed{100.0}$	$\boxed{100.0}$	98.5	$\boxed{98.7\pm1.0}$	97.2	$\boxed{98.4\pm1.3}$

C. Experimental Results

We did the experiments for steps $s = 1, 2, 3, 4$ and 5. To shorten the paper, we show in Table I only the results obtained with step $s = 2$ that yielded the highest key recognition rate. A box encloses the best results for each scheme. The rates depicted in this table are the averages of the 7 tests, leaving one group out each time. “Dim” indicates the number of features used in the experiment, where Dim = $3 \times (2M + 1)$. The number after ± is the standard deviation.

Surprisingly, we achieved 100% of row recognition rate and 98.4% of key recognition rate.

These rates are unexpectedly high, even considering that the results were obtained using a laboratory test fixture.

Note that the row scheme yields higher recognition rates than the column scheme. The opposite will be observed in PIN experiments. This fact can be explained considering that the acrylic plate where the ATM keypad was assembled is larger in width (x) than in depth (y), as can be seen in Fig. 2. So, the moment of inertia around $\vec{x}$ axis is smaller than the moment around $\vec{y}$ axis. Consequently, it is easier to shake or rotate the equipment around $\vec{x}$ than around $\vec{y}$.

V. PIN1: The Preliminary PIN-pad Experiment

A. Experiment Assembly

We made two PIN-pad experiments using Tecvan/Gertec PPC800 PIN-pad with standard matrix of keys (Fig. 4) and two accelerometers (Fig. 5). In both experiments, we enveloped two digital accelerometers in heat shrink tubes and glued them in the positions depicted in Fig. 5.

The accelerometers do not have any electrical connection with the PIN-pad. After implanting the “bugs,” the bottom cover was put back.

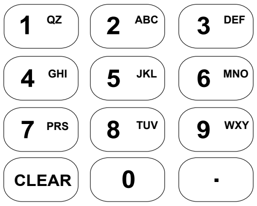

Fig. 4. Keyboard layout of the PIN-pad terminal.

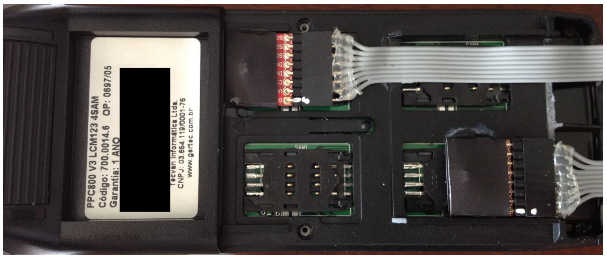
[redacted]

Fig. 5. The compartment at the bottom of the PPC800 PIN-pad, with the two accelerometers glued over two SAM (Security Authentication Module) slots.

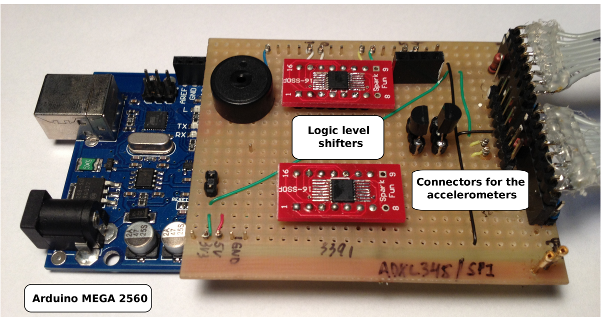

Fig. 6. An Arduino board was used as the acquisition system. A supplementary board containing auxiliary circuits and connectors for the accelerometers was assembled over it.

In preliminary PIN1 experiment, we used Freescale MMA8452 digital low-g accelerometers, with 12 bits resolution and three axes [22]. We used an Arduino Mega [23] board as the acquisition system (Fig. 6). Based on the previous experiment, we considered adequate the acceleration range of ±2g for keystroke measurement purposes. We used the maximum sampling rate allowed by the accelerometers, 800 samples/s. For each keystroke, 300 samples from all three axes of both sensors were acquired, totalizing 1800 samples. The communication, storage and further data processing were carried out on a PC.

TABLE II

Recognition rate in % obtained in PIN 1 experiment, “rigid mode.” Step s = 2

		Row Scheme		Column Scheme		Key Scheme	
Feature	Dim	MLP	SVM	MLP	SVM	MLP	SVM
same	45	33.8	37.2	64.3	70.5	24.4	30.1
	63	32.1	37.8	64.1	72.5	24.5	30.2
comb	45	38.4	44.5	78.6	81.6	34.3	38.5
	75	40.0	$\boxed{46.5\pm3.0}$	80.7	$\boxed{84.2\pm5.6}$	37.2	$\boxed{42.1\pm5.2}$

B. Data Acquisition

We captured the keystroke samples from 3 people in a total of 7 sessions (4 sessions from the first person, 2 from the second and one session from the third). In each session, a person pressed 40 times each of the 12 keys, totalizing 3360 keystrokes. The data produced by 6 out of 7 sessions were used for the training. We reserved the data from the remaining session for the testing, using the “leave-one-out” cross-validation method. In this experiment, the keys were pressed with the PIN-pad resting over a rigid table.

C. Experimental Results

The recognition rates depicted in Table II are the average of the 7 rates, using step $s = 1$. “Dim” indicates the number of features used in the experiment, where Dim = $3 \times (2M + 1)$ for “same” features and Dim = $15 \times (2M + 1)$ for “comb” features.

In this case, the column recognition rates are much higher than the row recognition rates. In all schemes, the combination of $\mathbf{comb^{PIN}}$ feature with dimension 75 and SVM yielded the highest recognition rates. The key recognition rate of 42.1% is disappointing, compared with 98.4% achieved in ATM experiment. So, we carried out the “improved PIN-pad experiment,” described in the next Section, in an attempt to raise it.

VI. PIN2: The Improved PIN-pad Experiment

A. Experiment Assembly

PIN2 experiment assembly was very similar to PIN1’s. We used the same PIN-pad but chose another model of accelerometers: Analog Devices ADXL345 digital low-g accelerometers, with 10 bits resolution, three axes and sampling rate up to 3200 samples/s [25]. We chose this model because we considered that the low sampling rate of PIN1 experiment (800 samples/s in contrast to 6700 samples/s in ATM experiment) may be responsible for the low recognition rates. So, we used the maximum sampling rate allowed by the new accelerometers.

B. Data Acquisition

We captured the keystroke samples from 5 people, each person participating in 2 sessions. In the first session, the person pressed the keys with the PIN-pad resting on a rigid surface. In the second session, the person pressed the keys holding the PIN-pad in hand. Each session was composed of 40 keystrokes for each of the 12 keys of the keyboard depicted in Fig. 4. A total of 4800 keystrokes were acquired (5 people $\times$ 2 sessions $\times$ 12 keys $\times$ 40 keystrokes). For each keystroke, 300 acceleration values from the three axes of both sensors were acquired, totalizing 1800 samples. We used the session 1 data ("rigid mode") independently from the session 2 data ("hold mode"). We used the data produced by 4 out of 5 people for training and the remaining data for testing. The results presented in the tables are the averages of the 5 results, leaving one person out each time.

C. Sample Alignment

We observed that the sampling rates of the two accelerometers were slightly different, since the digital accelerometers have independent internal clocks. Also, the acquisition system alternates the samplings between the two sensors, i.e., samples from the two sensors are not acquired at the same time, causing a temporal misalignment in the acquired data. So, in order to minimize the clock synchronization problem, we stored the timestamp of each acceleration sample acquired. Then, we performed a cubic spline interpolation to obtain samples evenly distributed in time. We compared the results with and without alignment.

D. Experimental Results of "Rigid Mode"

Table III shows the recognition rates with the PIN-pad resting on a rigid surface, without time alignment. As in ATM experiment, we did the calculations for steps $s = 1, 2, 3, 4$ and 5. However, to shorten the paper, we show only the rates obtained with step $s = 2$.

The recognition rates are significantly higher than those obtained in PIN1 experiment (Table II), probably due to the increased sampling rate (3200 samples/s instead of 800 samples/s). The key recognition rate improved from 42.1% (Table II) to 67.1% (Table III), using Dim=75. The highest key recognition rate is 68.6%, using Dim=165.

TABLE III

Recognition rate in % obtained in PIN 2 experiment, "rigid mode" without time alignment. Step s = 2

		Row Scheme		Column Scheme		Key Scheme	
Feature	Dim	MLP	SVM	MLP	SVM	MLP	SVM
same	45	57.4	58.4	84.0	86.1	53.9	55.8
	63	54.7	60.3	84.4	87.2	54.0	55.6
comb	45	61.2	62.1	86.8	89.3	59.7	61.1
	75	64.3	63.7	90.0	91.5	63.8	67.1
	105	66.0	64.4	91.1	91.4	66.0	68.1
	135	65.7	70.6	92.8	93.3	67.4	67.6
	165	68.5	$\boxed{71.3 \pm 4.3}$	93.8	$\boxed{94.1 \pm 2.8}$	68.3	$\boxed{68.6 \pm 7.1}$

TABLE IV

Recognition rate in % obtained in PIN 2 experiment, "rigid mode" with time alignment. Step s = 2

		Row Scheme		Column Scheme		Key Scheme	
Feature	Dim	MLP	SVM	MLP	SVM	MLP	SVM
same	45	66.7	68.9	85.6	88.7	68.6	67.2
	63	66.5	68.6	88.2	89.7	67.2	69.3
comb	45	65.6	68.1	88.4	90.1	65.8	65.2
	75	70.1	69.9	91.9	91.6	71.2	71.5
	105	72.1	72.6	92.4	93.9	74.1	74.1
	135	73.5	76.0	93.4	94.4	76.1	74.5
	165	74.6	$\boxed{77.1 \pm 3.3}$	94.8	$\boxed{95.5 \pm 2.7}$	$\boxed{76.7 \pm 5.5}$	75.7

Table IV shows the recognition rates for the same data with sampling time alignment, for step $s = 2$. The recognition rates improved again. With time alignment, the best key recognition rate increased from 68.6% to 76.7%, this time using "comb" features with dimension 165.

E. Experimental Results of "Hold Mode"

Tables V and VI show the recognition rates with the PIN-pad hold in hand, respectively without and with time alignment, for step $s = 2$. In general, the recognition rates in "hold mode" are higher than the corresponding ones in "rigid mode." Searching for a rational explanation for this fact, we computed the average periodogram[2] of acceleration data in $\vec{z}$ axis for all the data of "rigid" and "hold" modes (Fig. 7). There are much more low-frequency power (5-40 Hz) in "hold mode" than in "rigid mode." We

[2] We used MATLAB 7.0.8 function "periodogram" to compute it.

TABLE V

Recognition rate in % obtained in PIN 2 experiment, “hold mode” without time alignment. Step s = 2

		Row Scheme		Column Scheme		Key Scheme	
Feature	Dim	MLP	SVM	MLP	SVM	MLP	SVM
same	45	73.9	71.4	78.0	78.7	65.4	62.4
	63	74.6	73.0	78.9	77.3	64.4	62.7
comb	45	66.6	63.2	89.0	89.9	64.0	65.3
	75	70.4	69.0	93.6	92.0	70.3	67.7
	105	71.8	73.5	$\boxed{96.8\pm2.6}$	96.4	76.3	69.4
	135	74.8	74.8	96.5	96.5	$\boxed{78.5\pm3.9}$	72.7
	165	$\boxed{77.3\pm3.0}$	74.6	96.2	96.2	77.4	71.7

TABLE VI

Recognition rate in % obtained in PIN 2 experiment, “hold mode” with time alignment. Step s = 2

		Row Scheme		Column Scheme		Key Scheme	
Feature	Dim	MLP	SVM	MLP	SVM	MLP	SVM
same	45	$\boxed{82.0\pm3.1}$	78.3	80.1	79.9	73.8	69.8
	63	81.1	79.9	83.4	79.7	75.0	68.8
comb	45	73.8	73.1	89.6	90.9	71.8	69.4
	75	77.2	76.2	94.6	93.1	76.5	70.6
	105	80.7	77.9	96.8	96.2	80.3	73.5
	135	81.6	78.2	96.5	96.5	$\boxed{82.1\pm3.6}$	75.6
	165	81.9	78.1	96.5	$\boxed{97.1\pm2.9}$	81.7	76.6

hypothesize that these low-frequency components have important information that help to identify the pressed key. Maybe the additional vibrations introduced by the user holding the PIN-pad may also have influenced to improve the detection accuracy.

On the other hand, “rigid mode” has more power around 100 Hz than “hold mode,” maybe because hand absorbs vibrations in this frequency.

The recognition rates with time alignment (Table VI) are higher than the corresponding ones without time alignment (Table V). This demonstrates that the time alignment improves the quality of the obtained data, and it is an essential preprocessing task in this scenario. The best key recognition rate of all PIN-pad experiments was 82.1%, obtained in “hold mode” with time alignment, using “comb” feature with dimension 135 combined with MLP (Table VI).

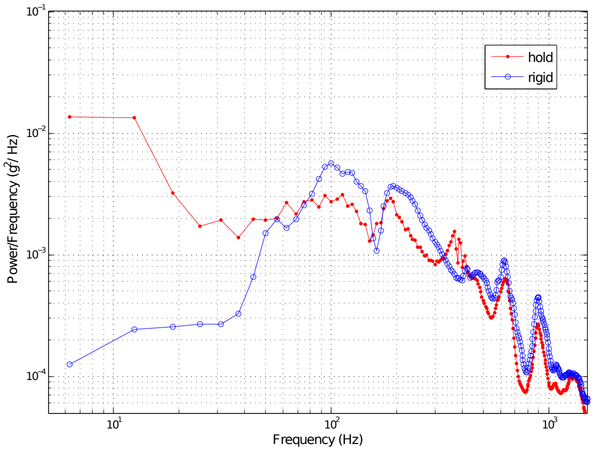

Fig. 7. Average periodogram of $\vec{z}$ axis for "hold mode" and "rigid mode"

TABLE VII

PROCESSING TIMES IN MILLISECONDS OF ONE LEAVE-ONE-OUT SESSION USING $\mathbf{comb}$ FEATURE WITH DIMENSION 75.

	MLP	SVM
Learning	9377	59750
Classification	1.7	20

F. Processing Times

We present in Table VII the learning and classification times for one leave-one-out session using $\mathbf{comb}^{\mathbf{PIN}}$ feature with dimension 75 in a Intel® Xeon® 3.3GHz computer running Windows 7. We do not intend to make a detailed performance comparison between the algorithms, but only inform the reader the magnitude of the computational effort involved.

G. Considerations

PIN2 recognition rates are overall higher than PIN1 rates because:

1) We increased the sampling rate four times.
2) We used numerical interpolation to align the samples in time.

However, ATM recognition rates are even higher than PIN2's. In our opinion, this happens because:

1) In ATM experiment, the system reads the three acceleration values of the three sensors at once with negligible delay. The sampling rate was constant and provided by a single clock. In contrast, in PIN experiments, each sensor has an independent clock with slightly different sampling rates.

Additionally, the precise instant of sampling depends on the availability of the data bus shared by the two sensors.

2) ATM experiment used sampling rate of 6700 samples/s, twice PIN2's and 8 times PIN1's.
3) The use of three sensors instead of two may improve the key recognition due to geometric triangulation.

H. Row and Column Recognition

In all PIN-pad experiments, columns were always easier to identify than rows. This fact can be explained by the same rationale used in Section IV. The PIN-pad is elongated (Fig. 8), that is, it is larger in depth (y) than in width (x). So, the moment of inertia around I_y is smaller than around I_x, where I_y and I_x are the central inertial axes in directions $\vec{y}$ and $\vec{x}$. In the equipment we used, the moment of inertia around I_y is roughly 6 times smaller than around I_x. This means that it is much easier to shake the equipment around I_y than I_x, making it easier to identify columns than rows.

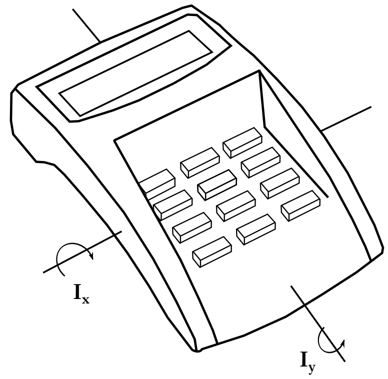

Fig. 8. It is easier to identify columns than rows because the moment of inertia around I_y is smaller than around I_x.

I. Weakness in PIN-pad Design

The design of most PIN-pads has a weakness that facilitates the mechanical vibration attack. These devices have tampering detection mechanisms that destroy sensitive information, such as cryptographic keys, in case of a tampering event. Still, the devices have a cover at their bottom to permit legitimate access to the SAM (Security Authentication Module) card slots. SAM cards are responsible for the secure communication between the terminal and the payment companies. The empty space under the cover can be used to install devices that illegally collect information, "bugs" in the security jargon. Moreover, the

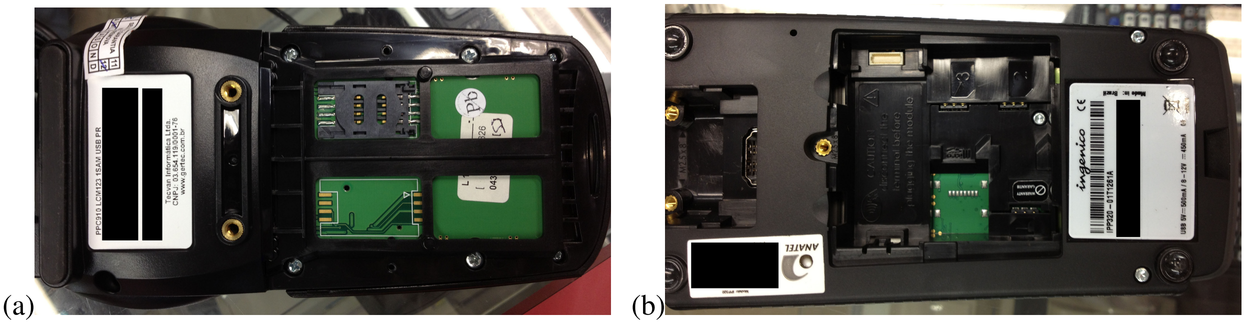

Fig. 9. Bottom views of two PIN-pads: (a) Tecvan/Gertec PPC910 and (b) Ingenico IPP320. Both have plenty of space to implant a "bug," making them potentially vulnerable to vibration attack.

SAM slots themselves can be used for the attack. First, they can be used as receptacles for fake SAM cards containing the accelerometers and some auxiliary circuitry (for example, wireless communication). Second, the SAM slots can provide electrical power for the "bugs." Thus, the mechanical vibration attack can be executed in real scenario in a noninvasive and undetectable way, without batteries and wires. We mean, by word "noninvasive," that there is no physical harm to the targeted device [27].

The PIN-pad we tested (Tecvan/Gertec PPC800) is not PCI-PTS[3] compliant. Nevertheless, compliant models such as PPC910 and Ingenico IPP320 present the same empty space at bottom where "bugs" can be implanted (Fig. 9). These are very recent models easily found in retail stores.

Note that in our tests we have actually implemented the "bugs" in the empty space under the bottom cover (the SAM compartment). Our system only lacks the wireless transmission capability to be able to make a real attack, because at this time it uses cables to communicate with the computer and also to receive the power. It is technically feasible and economically viable to make a wireless eavesdropping device, even in a home laboratory. See for instance the submillimeter thick EMV[4] card monitor in [26], where a full acquisition system for data eavesdropping (containing a microcontroller, a flash memory chip and some glue logic) was embedded in the body of a plastic card, for protocol analysis purpose. The researchers show that all the circuitry has only 0.81mm of thickness, ready to be inserted in a smart card reader slot. The SAM compartment usually provides much more free space.

[3] PTS stands for PIN Transaction Security, a set of requirements specific for PIN entry devices, proposed by the PCI. Device compliance can be consulted at https://www.pcisecuritystandards.org/approved_companies_providers/approved_pin_transaction_security.php

[4] EMV stands for Europay, MasterCard and Visa, a global standard for inter-operation of integrated circuit cards

VII. Conclusion

In this paper, we have addressed the problem of identifying the pressed keys in an ATM keypad and a PIN-pad by analyzing mechanical vibrations produced by the keystrokes. We have achieved key recognition rate of 98.4% in ATM experiment. This rate is extremely high, even considering that the result comes from a laboratory environment. It shows how well correlated are the keystrokes and the vibrations, leaking almost all information. We have also achieved key recognition rates of 76.7% and 82.1% using the PIN-pad resting on a rigid surface and hold in hand, respectively. These rates are also very high and certainly they all are security breaches. In information security, the disclosure of any data that increase the knowledge of the confidential information is considered a security breach. We have also considered that the easy access to the SAM card compartment increases the vulnerability of PIN-pads, permitting an undetectable installation of the accelerometers and communication systems. The described attack can be executed with very low budget, using straightforward techniques and open-source hardware and software.

References

[1] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi, "The EM side-channel(s)," in *Cryptographic Hardware and Embedded Systems*, LNCS, vol. 2523, pp. 29–45, 2003.

[2] F. -X. Standaert, T. Malkin, and M. Yung, "A unified framework for the analysis of side-channel key recovery attacks," in *Advances in Cryptology - EUROCRYPT*, LNCS, vol. 5479, pp. 443–461, 2009.

[3] M. G. Kuhn, "Compromising emanations: eavesdropping risks of computer displays," University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-577, 2003.

[4] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard acoustic emanations revisited," *ACM T. Information and System Security*, vol. 13, no. 1, pp. 3:1–3:26, Oct. 2009.

[5] Y. Berger, A. Wool, and A. Yeredor, "Dictionary attacks using keyboard acoustic emanations," in *Proc. 13th ACM Conf. Computer and Communications Security*, pp. 245–254, 2006.

[6] T. Halevi and N. Saxena, "A Closer Look at Keyboard Acoustic Emanations: Random Passwords, Typing Styles and Decoding Techniques." [Online]. Available: http://eprint.iacr.org/, 2010.

[7] D. Asonov and R. Agrawal, "Keyboard acoustic emanations," in *Proc. IEEE Symp. Security and Privacy*, pp. 3, 2004.

[8] M. Backes, M. Dörmuth, S. Gerling, M. Pinkal, and C. Sporleder, "Acoustic side-channel attacks on printers." in *Proc. USENIX Security Symposium*, pp. 307–322, 2010.

[9] T. Halevi, and N. Saxena, "On pairing constrained wireless devices based on secrecy of auxiliary channels: the case of acoustic eavesdropping," in *Proc. 17th ACM Conf. on Computer and communications security*, 2010.

[10] A. Shamir and E. Tromer, "Acoustic cryptanalysis - on nosy people and noisy machines." [Online]. Available: http://www.cs.tau.ac.il/~tromer/acoustic

[11] L. Cai and H. Chen, "Touchlogger: inferring keystrokes on touch screen from smartphone motion," in *Proc. 6th USENIX Conf. Hot Topics in Security*, 2011.

[12] E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R.R. Choudhury, “Tapprints: your finger taps have fingerprints,” in *Proc. 10th international Conf. on Mobile Systems, Applications, and Services*, 2012.

[13] L. Cai, S. Machiraju, and H. Chen, “Defending against sensor-sniffing attacks on mobile phones,” in *Proc. 1st ACM Workshop on Networking, Systems, and Applications for Mobile Handhelds*, pp. 31–36, 2009.

[14] Payment Card Industry - Security Standards Council LLC, *PIN Transaction Security (PTS) Point of Interaction (POI) Derived Test Requirements v3.1*, October 2011. [Online]. Available: https://www.pcisecuritystandards.org/documents/PCI_PTS_POI_DTRs_v3_1.pdf

[15] G.S. Faria and H.Y. Kim, “Identificação das teclas digitadas a partir da vibração mecânica” in *Anais do 30º Simpósio Brasileiro de Telecomunicações*, SBrT, Set. 2012.

[16] P. Marquardt, A. Verma, H. Carter, and P. Traynor, “(sp)iPhone: decoding vibrations from nearby keyboards using mobile phone accelerometers,” in *Proc. 18th ACM Conf. on Computer and Communications Security*, 2011.

[17] J. P. Lewis, “Fast template matching,” in *Proc. Vision Interface*, pp. 120–123, 1995.

[18] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, O’Reilly, 2008.

[19] R. P. Lippmann, “An introduction to computing with neural nets,” *IEEE ASSP Magazine*, vol. 4, no. 2, pp. 4 –22, Apr. 1987.

[20] J. A. K. Suykens and J. Vandewalle, “Least squares support vector machine classifiers,” *Neural Processing Letters*, vol. 9, pp. 293–300, 1999.

[21] Freescale Semiconductor - MMA7260QT: 3-Axis Acceleration Sensor. [Online]. Available: http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MMA7260QT

[22] Freescale Semiconductor - MMA8452: Xtrinsic 3-Axis, 12 Bit Accelerometer. [Online]. Available: http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MMA8452Q&fr=g

[23] Arduino Board Mega 2560. [Online]. Available: http://arduino.cc/en/Main/ArduinoBoardMega2560

[24] R. Kohavi, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Proc. of the 14th international joint Conf. on Artificial Intelligence*, vol. 2, pp. 1137–1143, 1995.

[25] Analog Devices - ADXL345 3-Axis, ±2g, ±4g, ±8g, ±16g Digital Accelerometer. [Online]. Available: http://www.analog.com/static/imported-files/data_sheets/ADXL345.pdf

[26] M. Bond, O. Choudary, S.J. Murdoch, S. Skorobogatov, and R. Anderson, “Chip and Skim: cloning EMV cards with the pre-play attack,” p. 8, Sept. 2012. [Online]. Available: http://www.cl.cam.ac.uk/~rja14/Papers/unattack.pdf

[27] R. Anderson, *Security Engineering: A Guide to Building Dependable Distributed Systems*, Wiley Publishing, ch. 16, 2008.

Gerson de Souza Faria received the B.Sc. degree in Physics from University of São Paulo (USP) in 2002 and M.Sc. in Electrical Engineering in 2012 from Escola Politécnica, USP. Currently, he is a Ph.D. candidate in Electrical Engineering in the same department. His interests include signal and image processing, machine learning, and information security in human-machine interfaces.

Hae Yong Kim was born in South Korea in 1964 and migrated to Brazil in 1975. He received the third highest score in the entrance exam to the University of São Paulo (USP), among about 11000 candidates to Sciences and Engineering, and has graduated in Computer Science in 1988 with the best average scores. He received M.Sc. in Applied Mathematics (1992) and Ph.D. in Electrical Engineering (1997), both from USP. He has lectured at USP since 1989, and is currently an associate professor with the Department of Electronic Systems Engineering, Escola Politécnica, USP. He has been receiving academic productivity scholarship from CNPq (National Counsel for Technological and Scientific Development) since 2002. His research interests include the general area of image and video processing, object recognition, authentication watermarking, information security and machine learning.