Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Side-channel attacks and learning-vector quantization

Ehsan SAEEDI[†‡], Yinan KONG, Md. Selim HOSSAIN

(*Department of Engineering, Macquarie University, Sydney, Australia*)

[†]E-mail: saeedi.ehsan@gmail.com

Received Dec. 19, 2015; Revision accepted Feb. 28, 2016; Crosschecked Mar. 28, 2017

Abstract: The security of cryptographic systems is a major concern for cryptosystem designers, even though cryptography algorithms have been improved. Side-channel attacks, by taking advantage of physical vulnerabilities of cryptosystems, aim to gain secret information. Several approaches have been proposed to analyze side-channel information, among which machine learning is known as a promising method. Machine learning in terms of neural networks learns the signature (power consumption and electromagnetic emission) of an instruction, and then recognizes it automatically. In this paper, a novel experimental investigation was conducted on field-programmable gate array (FPGA) implementation of elliptic curve cryptography (ECC), to explore the efficiency of side-channel information characterization based on a learning vector quantization (LVQ) neural network. The main characteristics of LVQ as a multi-class classifier are that it has the ability to learn complex non-linear input-output relationships, use sequential training procedures, and adapt to the data. Experimental results show the performance of multi-class classification based on LVQ as a powerful and promising approach of side-channel data characterization.

Key words: Side-channel attacks; Elliptic curve cryptography; Multi-class classification; Learning vector quantization

http://dx.doi.org/10.1631/FITEE.1500460 **CLC number:** TP309

1 Introduction

Over the past decade, there has been a dramatic increase in various applications and implementations of side-channel attacks (SCAs). Because SCAs can generally be performed using relatively cheap equipment, they pose a serious threat to the security of most cryptographic hardware devices. Such devices range from personal computers to small embedded devices, such as smart cards and radio frequency identification devices (RFIDs). However, different from the protocols that have been used as secure methods of data communication, SCAs try to highlight their vulnerabilities. For instance, two-factor authentication is proposed as a simple, portable, and robust protocol; however, a few studies (Wang and Wang, 2015; Wang *et al.*, 2015a; 2015b) have highlighted its vulnerabilities and enhanced its security at nearly no additional communication or computation cost.

Yeh (2015) introduced a novel authentication protocol for security enhancement and for eliminating the weaknesses of previous works. Li and Lee (2011) developed a secure scheme and claimed that it is secure against the smart-card-loss attack. Wang *et al.* (2013) presented a robust scheme to cope with the defects of Li and Lee (2011)'s scheme, while retaining the merits of different password authentication schemes using smart cards. Furthermore, an improved dynamic ID-based authentication scheme was proposed to remedy previous security flaws. Ma *et al.* (2012; 2014) presented three principles that are helpful in explaining many of the security failures repeated in the past and important for designing more robust schemes in the future. Recently, a considerable body of literature has grown up around the theme of side-channel information analysis

‡ Corresponding author
ORCID: Ehsan SAEEDI, http://orcid.org/0000-0002-0879-113X

methods. First, it was shown as simple power analysis (SPA) and simple electromagnetic analysis (SEMA), which rely on pattern recognition in a power or electromagnetic signal trace. Afterwards, differential power analysis (DPA) and differential electromagnetic analysis (DEMA) were introduced, and in advanced statistical algorithms they were used to theoretically analyze the signal traces (Kocher *et al.*, 1999; de Mulder *et al.*, 2005). In addition, several attempts have been made to exploit side-channel information through a profiling-based attack, in which a training device that is fully controllable and accessible is used in the training phase to gain additional knowledge for the attack against an identical target device. Saeedi and Kong (2014) introduced machine learning as a powerful type of profiling-based SCA from an information theoretical point of view. A number of studies (Heuser and Zohner, 2012; Bartkewitz and Lemke-Rust, 2013; Saeedi and Kong, 2014; Saeedi *et al.*, 2015) have used support vector machines (SVMs) as powerful classifiers to classify different patterns of side-channel information. More recent studies have confirmed that neural networks have led to the emergence of powerful tools in solving classification and pattern recognition problems, and they can be considered promising alternatives to various conventional classification methods (Cybenko, 1989; Haykin, 2009).

Numerous studies have attempted to address this issue of countermeasures against conventional SCAs. Most of the approaches are based on implementations of a cryptography algorithm with constant or randomized execution time or execution order (also known as 'shuffling') to render the occurrence of unpredictable leakage (Mangard *et al.*, 2007; Tillich and Herbst, 2008).

While several countermeasures against conventional attacks have been proposed, cryptosystems are still vulnerable to SCAs because some inherent leakages during single executions in a cryptography algorithm cannot be prevented in many cases, e.g., location-based leakage (Heyszl *et al.*, 2012a), address bit leakage (Itoh *et al.*, 2002), or operation-dependent leakage (Prouff, 2014). Furthermore, most of the countermeasures have a negative effect, sometimes significant, on the performance of cryptosystems (Kopf and Durmuth, 2009) or cost of implementation (Tillich and Herbst, 2008).

To the best of our knowledge, there are only a few studies that address SCAs based on a neural network, and no attempt has been made to explore the performance of a learning vector quantization (LVQ) neural network in analyzing side-channel information. In addition, both attacks and countermeasures interact strongly and, while the adversary needs only to succeed with one out of many attack methods, the designers have to consider all the known attacks, whenever applicable to their system, simultaneously. Thus, the verification of the new techniques of SCAs plays a crucial role for cryptosystem designers. In this paper, neural networks are applied as a powerful and efficient method for the characterization of side-channel information. To classify side-channel information, a multi-class classifier based on LVQ neural networks is used. Our experimental investigation is aimed to verify the performance of classification for the different training algorithms and different numbers of hidden layers. The experiment is performed with a filed-programmable gate array (FPGA) implementation of elliptic-curve cryptography (ECC). ECC is one of the most common public-key cryptography methods because of its major benefits relative to other algorithms; namely, it has more security per bit and a suitable key size for hardware. In this work, we implement all elliptic-curve operations in an affine coordinate system. We present the EC scalar multiplication left-to-right algorithm using the binary method, which is implemented by the double-and-add algorithm. For details on ECC implementation and considerations, the reader is referred to other studies (Miller, 1986; Koblitz, 1987; Blake *et al.*, 1999).

2 Neural networks as multi-class classifiers

Neural networks are powerful tools for classification and pattern recognition tasks and are typically organized in layers. Layers are made up of a number of interconnected 'nodes' that contain an 'activation function'. Patterns are presented to the network via the 'input layer', which communicates to one or more 'hidden layers' where the actual processing is done via a system of weighted 'connections'. The hidden layers then link to an 'output layer' where the answer is output. Most neural networks contain some form of 'learning rule', which modifies the weights of the connections according to the input patterns that it

is presented with. The learning process involves updating network architecture and connection weights so that a network can efficiently perform a specific classification/clustering task.

2.1 Side-channel attacks based on neural networks

A neural network learns the signature (power consumption and electromagnetic emission) of an instruction, and then recognizes it automatically. For each instruction, hundreds of structures need to be stored for a cryptosystem processor. Modeling the power leakage is considered as the basis for launching SCAs, and the effectiveness of these attacks strongly depends on the accuracy of the underlying side-channel leakage characterization. The general goal of an attack is to obtain the secret key value, stored in the cryptographic module, from the measured power trace. Considering the value K_{sec} as a secret key stored in the attacked cryptographic module and K_{est} as the estimate value of the secret key, determined with a neural network, if the method works correctly, the values of K_{est} and K_{sec} will be equal at the end of the classification process.

3 Multi-class classification based on learning vector quantization

LVQ is a popular classifier for multi-class classification. Considering simplicity, classification accuracy, and training speed, the LVQ classifier compares favorably to other classification methods and has successfully been applied in many pattern recognition domains, e.g., speech recognition (Mäntysalo *et al.*, 1992) and radar classification (Orlando *et al.*, 1990). Flotzinger *et al.* (1992) applied LVQ to the classification of electroencephalogram patterns.

Vector quantization (VQ) has been extensively explored from theoretical and empirical points of view. There are a couple of classical reviews on this topic (Gersho, 1979; Zador, 1982). A vector quantizer maps k-dimensional vectors in the vector space $\mathbb{R}^k$ into a finite set of vectors $Y = \{\boldsymbol{y}_i : i = 1, 2, \cdots, N\}$. Each vector $\boldsymbol{y}_i$ is called a code vector or a codeword, and the set of all the codewords is called a codebook. Associated with each codeword $\boldsymbol{y}_i$ is a nearest-neighbour region called the Voronoi region, defined by

$$V_i = \{\boldsymbol{x} \in \mathbb{R}^k : ||\boldsymbol{x} - \boldsymbol{y}_i|| \leq ||\boldsymbol{x} - \boldsymbol{y}_j||, \ \forall\, j \neq i\}. \quad (1)$$

The set of Voronoi regions partitions the entire space $\mathbb{R}^k$ such that

$$\bigcup_{i=1}^{N} V_i = \mathbb{R}^k, \quad \bigcap_{i=1}^{N} V_i = \varnothing.$$

The main idea is to cover the input space of samples with code-book vectors (CVs), each representing a region labeled with a class. A CV can be seen as a prototype of a class member, localized in the center of a class region in the input space. A class can be represented by an arbitrary number of CVs, but one CV represents one class only (Fig. 1).

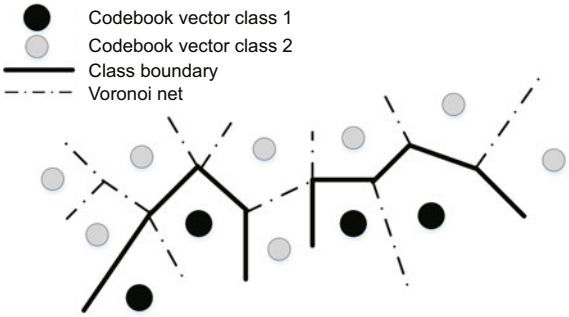

Fig. 1 Classification of the input space into class regions by codebook vectors in a two-dimensional feature space

Moving into a supervised context, LVQ (Kohonen, 1988) has a very important role in statistical pattern classification (Pregenzer *et al.*, 1996; Duda *et al.*, 2011). LVQ is a learning algorithm that combines competitive learning with supervision.

In terms of neural networks, an LVQ is a feedforward net with a two-layer neural network, including a competitive layer and a linear layer. The competitive layer is the core layer that performs classification through learning. Each neuron in the competitive layer of the LVQ network learns to recognize a prototype vector, which allows it to classify a region of the input space. By using LVQ networks, the distances between the input vectors and the prototype vectors are directly calculated. If two input vectors are close to each other, they belong to the same class. LVQ algorithms do not approximate the density functions of class samples, as VQ or probabilistic neural networks (PNNs) do, but directly define class boundaries based on prototypes. Fig. 1 shows the

classification of the input space into class regions by CVs represented as neurons positioned in a two-dimensional feature space. The LVQ architecture in a neural network is shown in Fig. 2.

The main advantages of LVQs are that they have the ability to learn complex non-linear input-output relationships, use sequential training procedures, and adapt themselves to the data. They can approximate any function with arbitrary accuracy. However, a major disadvantage of the LVQ classifier is that the success of a classification scheme may be directly associated with an appropriate data preprocessing transformation to normalize data and discard non-relevant input features. In addition, it is known to be a slow approach, which can affect its efficiency for real-time attacks.

3.1 Learning vector quantization algorithm

The basic LVQ algorithm, LVQ1, rewards correct classifications by moving the CV toward a presented input vector, whereas incorrect classifications are punished by moving the CV in the opposite direction. The magnitudes of these weight adjustments are controlled by a learning rate that can be lowered over time to obtain finer movements in a later learning phase. Improved versions of LVQ1 are KOHONEN's LVQ1 with different learning rates for each CV to obtain faster convergence, as well as LVQ2, LVQ2.1, and LVQ3.

A brief description of the most advanced training algorithm, LVQ3, is given below. Detailed descriptions of the currently available training algorithms can be found in recent studies (Kohonen, 1990a; 1990b; Pregenzer *et al.*, 1996).

Step 1: In an LVQ3 training iteration, $\boldsymbol{m}_i(t)$ and $\boldsymbol{m}_j(t)$ are the two codebook vectors closest to the present training sample $\boldsymbol{x}(t)$.

Step 2: Determine a symmetric 'window' of nonzero width around the mid-plane of $\boldsymbol{m}_i$ and $\boldsymbol{m}_j$. The condition in which a vector $\boldsymbol{x}$ can be defined to lie in the 'window' is

$$\min\left(\frac{d_i}{d_j},\frac{d_j}{d_i}\right)>s,$$

where d_i and d_j are the distances of $\boldsymbol{x}$ to $\boldsymbol{m}_i$ and $\boldsymbol{m}_j$, respectively, and s represents a constant factor, commonly chosen between 0.4 and 0.8.

Step 3: Update $\boldsymbol{m}_i$ and $\boldsymbol{m}_j$ by the LVQ3 training procedure with the following equations:

$$\begin{cases}\boldsymbol{m}_i(t+1)=\boldsymbol{m}_i(t)-\alpha(t)\left[\boldsymbol{x}(t)-\boldsymbol{m}_i(t)\right],\\ \boldsymbol{m}_j(t+1)=\boldsymbol{m}_j(t)+\alpha(t)\left[\boldsymbol{x}(t)-\boldsymbol{m}_j(t)\right],\end{cases}\tag{2}$$

if $\boldsymbol{x}$ falls into the 'window', $\boldsymbol{x}$ and $\boldsymbol{m}_j$ belong to the same class, while $\boldsymbol{x}$ and $\boldsymbol{m}_i$ belong to different classes;

$$\boldsymbol{m}_k(t+1)=\boldsymbol{m}_k(t)+\epsilon\alpha(t)\left[\boldsymbol{x}(t)-\boldsymbol{m}_k(t)\right],k\in\{i,j\},\tag{3}$$

if $\boldsymbol{x}$ falls into the 'window' and $\boldsymbol{x}$, $\boldsymbol{m}_j$, and $\boldsymbol{m}_i$ belong to the same class. $\alpha(t)$ is a scalar, decreasing monotonically in time. A common initial value $\alpha(0)$ is 0.03. ϵ is a constant; applicable values are between 0.1 and 0.5 (Kohonen, 1990a).

4 Experimental results based on learning vector quantization

This section is dedicated to the details of our experimental setup and results.

4.1 Experimental setup

In this experiment, the power consumption of an ECC cryptosystem is considered to be side-channel leakage. Fig. 3 shows the main measurement setup. As can be seen, the ECC cryptosystem is implemented on an FPGA board with a SPARTAN 3 FPGA. To record and see power-signal traces, a Tektronix TDS2012 oscilloscope with 1×10^9 samples/s is applied. In addition, to measure the power consumption and electromagnetic emission of our FPGA, a Tektronix CT1 current probe and an ETS near-field probe set (model 7405) are used, as well as an ETS broadband amplifier (model 7405-907b), to enhance the quality of the input signal traces. This experiment is performed using a MATLAB toolbox and a PC configuration of Intel Core i5, 2.80 GHz CPU, and 4.00 GB RAM.

4.2 Empirical results and discussions

Concerning our LVQ-based analysis, the number of hidden layers plays an important role in the overall neural network architecture and has a significant influence on the final output. There is no specific method or formula to determine this number, and hence this number must be carefully chosen via experiments. Using too few neurons can lead to

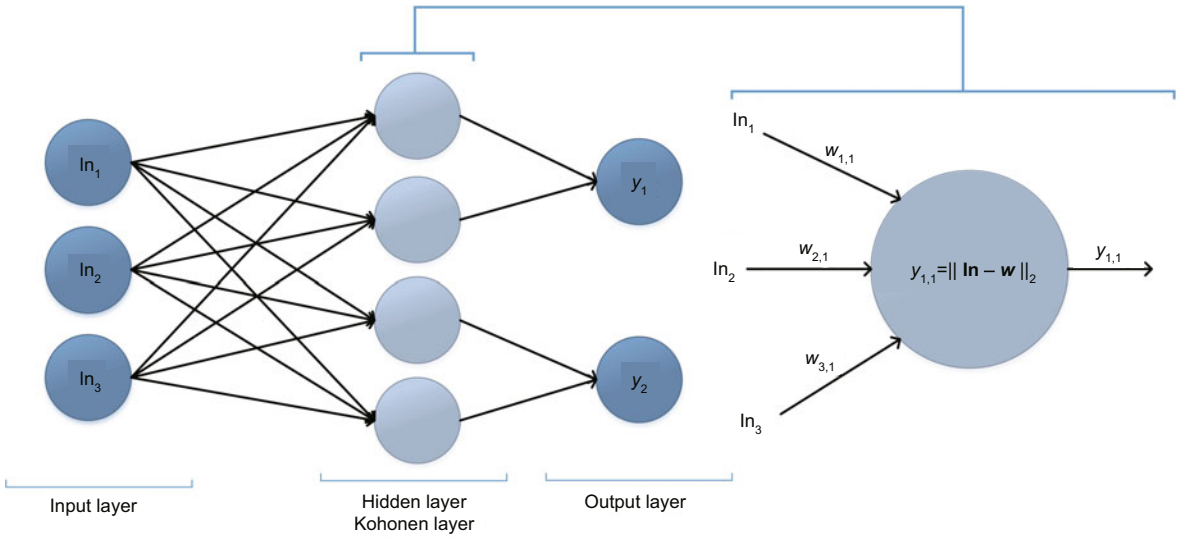

Fig. 2 LVQ architecture: one hidden layer with Kohonen neurons, adjustable weights between the input and hidden layer, and a winner-takes-all mechanism

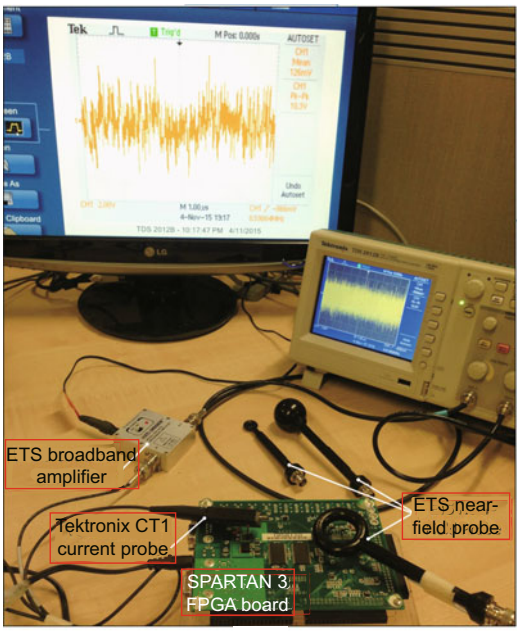

Fig. 3 Measurement setup for side-channel attacks

under fitting, while too many neurons can result in large computational complexity or over-fitting.

For this purpose, a comparison of classification accuracy, time complexity, and memory consumption between LVQ-based architectures with different numbers of hidden layers ranging from 10 to 110 is performed, and the experimental results are provided in Table 1. By increasing the number of hidden layers from 10 to 80, the mean squared error (MSE) drops from 0.135 to 0.060 and reaches a minimum of 0.057 with the number of hidden layers set at 90–100; after that, the error increases by 0.013 with 110 hidden layers.

Concerning time complexity, the most time-consuming LVQ architectures are those with the number of hidden layers between 90 and 100, with 14 000–16 000 s. The processing time increases gradually from 2303 to 12 500 s as the number of hidden layers increases from 10 to 80.

Judging from the memory consumption information in this table, the memory consumptions of all hidden layers are almost the same (in the range of $[0.124, 0.129]$).

Table 1 Learning vector quantization (LVQ) network performance comparison with a proper number of hidden layers and based on time complexity and memory consumption

Number of hidden layers	Mean squared error	Time (s)	Memory (GB)
10	0.135	2303	0.124
20	0.112	3253	0.124
30	0.090	4932	0.124
40	0.080	5841	0.125
50	0.070	6909	0.125
60	0.065	9639	0.129
70	0.060	11 247	0.128
80	0.062	12 512	0.127
90	0.057	13 915	0.129
100	0.057	16 023	0.128
110	0.070	5985	0.126

Fig. 4 shows the training performance of our LVQ-based classification. From this figure, the best training performance is 0.066556 at epoch 279.

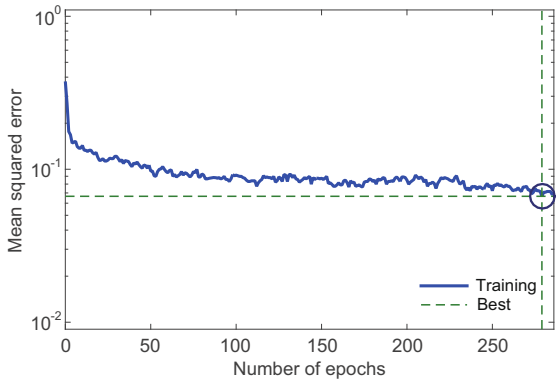

Fig. 4 The training performance of our learning vector quantization (LVQ) based classification

The efficiency of LVQ-based analysis is verified through a confusion matrix in which the number of times that a particular key value is correctly classified or misclassified is presented (Fig. 5). The diagonal cells show the number of times that the key values are correctly classified, and the off-diagonal cells show the misclassified cases. The blue cell in the bottom right shows the total percentage of correctly classified cases (in green) and the total percentage of misclassified cases (in red). As can be seen, the results show a good classification because of the higher numbers of the correct responses in the diagonal squares (143, 128, 110, and 136) compared to the relatively low numbers of incorrect responses in the off-diagonal squares ([0, 23]). The lower-right blue square illustrates the overall accuracies of 86.7% correctly classified and 14.3% misclassified.

Fig. 6 illustrates the receiver operating characteristic (ROC) curves to check the quality of classifiers. For each class of a classifier (classes of key bits 1, 2, 3, and 4), the threshold values are applied across the interval [0, 1] to outputs. For each threshold, two values are calculated: true positive ratio (the number of outputs greater than or equal to the threshold, divided by the number of '1' targets), and false positive ratio (the number of outputs less than the threshold, divided by the number of '0' targets). In Fig. 6, the colored lines represent the ROC curves, which are the plots of the true positive rate versus the false positive rate as the threshold is varied. A perfect test would show points in the upper-left corner. From this figure and considering the ratio of the true positive rate to the false positive rate, the best classification is for key bit 1, then for key bits 4, 2, and 3, in decreasing order.

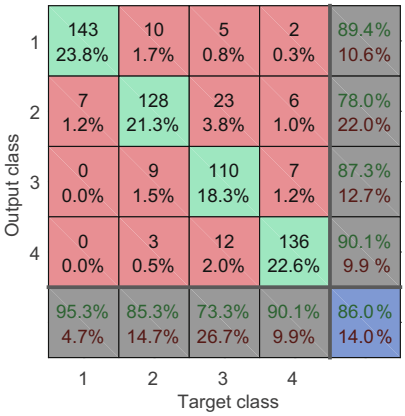

Fig. 5 Learning vector quantization (LVQ) based confusion matrix. Diagonal cells (green): number of correctly classified cases; off-diagonal cells (red): number of misclassified cases; blue cell: total percentages. References to color refer to the online version of this figure

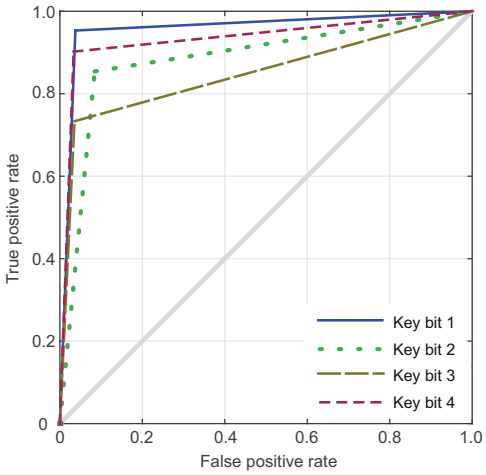

Fig. 6 Receiver operating characteristic (ROC) curves. The best classification is for key bit 1, then for key bits 4, 2, and 3, in decreasing order

To the best of our knowledge, there has been no attempt to explore the performance of an LVQ neural network in analyzing side-channel information. Nevertheless, in comparison with similar work (Heyszl *et al.*, 2012b), the maximum classification success rate achieved prior to our work was around 70%, apart from a study (Msgna *et al.*, 2014) that achieved a 100% classification success rate using a

specific combination of dimensionality reduction and a classification algorithm in an attack on an AVR processor (ATMega163).

5 Conclusions

In this paper, the characterization of side-channel information based on an LVQ neural network has been investigated. Considering our experimental results (based on an FPGA implementation of ECC), it is inferred that LVQ architectures with 90–100 hidden layers can be considered to be the most accurate architectures, although they are known to be the slowest. In addition, judging from the confusion matrices and the error histogram, our results indicate an overall accuracy of 86% correctly classified, 14% misclassified, and no significant over-fitting; therefore, LVQ can be considered a promising approach for side-channel data characterization.

References

Bartkewitz, T., Lemke-Rust, K., 2013. Efficient template attacks based on probabilistic multi-class support vector machines. *LNCS*, **7771**:263-276. http://dx.doi.org/10.1007/978-3-642-37288-9_18

Blake, I.F., Seroussi, G., Smart, N., 1999. Elliptic Curves in Cryptography. Cambridge University Press. http://dx.doi.org/10.1017/CBO9781107360211

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. *Math. Contr. Signals Syst.*, **2**(4):303-314. http://dx.doi.org/10.1007/BF02551274

de Mulder, E., Buysschaert, P., Ors, S.B., *et al.*, 2005. Electromagnetic analysis attack on an FPGA implementation of an elliptic curve cryptosystem. Int. Conf. on Computer as a Tool, p.1879-1882. http://dx.doi.org/10.1109/EURCON.2005.1630348

Duda, R.O., Hart, P.E., Stork, D.G., 2011. Pattern Classification. John Wiley & Sons.

Flotzinger, D., Kalcher, J., Pfurtscheller, G., 1992. EEG classification by learning vector quantization. *Biomed. Eng.*, **37**(12):303-309 (in German). http://dx.doi.org/10.1515/bmte.1992.37.12.303

Gersho, A., 1979. Asymptotically optimal block quantization. *IEEE Trans. Inform. Theory*, **25**(4):373-380. http://dx.doi.org/10.1109/TIT.1979.1056067

Haykin, S.S., 2009. Neural Networks and Learning Machines. Pearson Education, Upper Saddle River.

Heuser, A., Zohner, M., 2012. Intelligent machine homicide. Int. Workshop on Constructive Side-Channel Analysis and Secure Design, p.249-264. http://dx.doi.org/10.1007/978-3-642-29912-4_18

Heyszl, J., Mangard, S., Heinz, B., *et al.*, 2012a. Localized electromagnetic analysis of cryptographic implementations. Cryptographers' Track at the RSA Conf., p.231-244. http://dx.doi.org/10.1007/978-3-642-27954-6_15

Heyszl, J., Merli, D., Heinz, B., *et al.*, 2012b. Strengths and limitations of high-resolution electromagnetic field measurements for side-channel analysis. Int. Conf. on Smart Card Research and Advanced Applications, p.248-262. http://dx.doi.org/10.1007/978-3-642-37288-9_17

Itoh, K., Izu, T., Takenaka, M., 2002. Address-bit differential power analysis of cryptographic schemes OK-ECDH and OK-ECDSA. *LNCS*, **2523**:129-143. http://dx.doi.org/10.1007/3-540-36400-5_11

Koblitz, N., 1987. Elliptic curve cryptosystems. *Math. Comput.*, **48**(177):203-209. http://dx.doi.org/10.1090/S0025-5718-1987-0866109-5

Kocher, P., Jaffe, J., Jun, B., 1999. Differential power analysis. Annual Int. Cryptology Conf., p.388-397. http://dx.doi.org/10.1007/3-540-48405-1_25

Kohonen, T., 1988. An introduction to neural computing. *Neur. Networks*, **1**(1):3-16. http://dx.doi.org/10.1016/0893-6080(88)90020-2

Kohonen, T., 1990a. Improved versions of learning vector quantization. Int. Joint Conf. on Neural Networks, p.545-550. http://dx.doi.org/10.1109/IJCNN.1990.137622

Kohonen, T., 1990b. Statistical pattern recognition revisited. *In*: Eckmiller, R. (Ed.), Advanced Neural Computers. North-Holland, Amsterdam, p.137-144. http://dx.doi.org/10.1016/B978-0-444-88400-8.50020-0

Kopf, B., Durmuth, M., 2009. A provably secure and efficient countermeasure against timing attacks. 22nd IEEE Computer Security Foundations Symp., p.324-335. http://dx.doi.org/10.1109/CSF.2009.21

Li, C., Lee, C., 2011. A robust remote user authentication scheme using smart card. *Inform. Technol. Contr.*, **40**(3):236-245. http://dx.doi.org/10.5755/j01.itc.40.3.632

Ma, C., Wang, D., Zhang, Q., 2012. Cryptanalysis and improvement of Sood *et al.*'s dynamic ID-based authentication scheme. Int. Conf. on Distributed Computing and Internet Technology, p.141-152. http://dx.doi.org/10.1007/978-3-642-28073-3_13

Ma, C., Wang, D., Zhao, S., 2014. Security flaws in two improved remote user authentication schemes using smart cards. *Int. J. Commun. Syst.*, **27**(10):2215-2227. http://dx.doi.org/10.1002/dac.2468

Mangard, S., Oswald, E., Popp, T., 2007. Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer Science & Business Media. http://dx.doi.org/10.1007/978-0-387-38162-6

Mäntysalo, J., Torkkolay, K., Kohonen, T., 1992. LVQ-based speech recognition with high-dimensional context vectors. Int. Conf. on Spoken Language Processing, p.539-542.

Miller, V.S., 1986. Use of elliptic curves in cryptography. Conf. on the Theory and Application of Cryptographic Techniques, p.417-426. http://dx.doi.org/10.1007/3-540-39799-X_31

Msgna, M., Markantonakis, K., Mayes, K., 2014. Precise instruction-level side channel profiling of embedded processors. Int. Conf. on Information Security Practice and Experience, p.129-143. http://dx.doi.org/10.1007/978-3-319-06320-1_11

Orlando, J., Mann, R., Haykin, S., 1990. Radar Classification of Sea-Ice Using Traditional and Neural Classifiers. Proc. Int. Joint Conf. on Neural Networks, II-263.

Pregenzer, M., Pfurtscheller, G., Flotzinger, D., 1996. Automated feature selection with a distinction sensitive learning vector quantizer. *Neurocomputing*, **11**(1):19-29. http://dx.doi.org/10.1016/0925-2312(94)00071-9

Prouff, E., 2014. Constructive Side-Channel Analysis and Secure Design. Springer Berlin Heidelberg. http://dx.doi.org/10.1007/978-3-319-10175-0

Saeedi, E., Kong, Y., 2014. Side channel information analysis based on machine learning. 8th Int. Conf. on Signal Processing and Communication Systems, p.1-7. http://dx.doi.org/10.1109/ICSPCS.2014.7021075

Saeedi, E., Hossain, M.S., Kong, Y., 2015. Multi-class SVMs analysis of side-channel information of elliptic curve cryptosystem. Int. Symp. on Performance Evaluation of Computer and Telecommunication Systems, p.1-6. http://dx.doi.org/10.1109/SPECTS.2015.7285297

Tillich, S., Herbst, C., 2008. Attacking state-of-the-art software countermeasures: a case study for AES. Int. Workshop on Cryptographic Hardware and Embedded Systems, p.228-243. http://dx.doi.org/10.1007/978-3-540-85053-3_15

Wang, D., Wang, P., 2015. Offline dictionary attack on password authentication schemes using smart cards. *LNCS*, **7807**:221-237. http://dx.doi.org/10.1007/978-3-319-27659-5_16

Wang, D., Ma, C., Zhang, Q., *et al.*, 2013. Secure password-based remote user authentication scheme against smart card security breach. *J. Networks*, **8**(1):148-155.

Wang, D., He, D., Wang, P., *et al.*, 2015a. Anonymous two-factor authentication in distributed systems: certain goals are beyond attainment. *IEEE Trans. Depend. Sec. Comput.*, **12**(4):428-442. http://dx.doi.org/10.1109/TDSC.2014.2355850

Wang, D., Wang, N., Wang, P., *et al.*, 2015b. Preserving privacy for free: efficient and provably secure two-factor authentication scheme with user anonymity. *Inform. Sci.*, **321**:162-178. http://dx.doi.org/10.1016/j.ins.2015.03.070

Yeh, K., 2015. A lightweight authentication scheme with user untraceability. *Front. Inform. Technol. Electron. Eng.*, **16**(4):259-271. http://dx.doi.org/10.1631/FITEE.1400232

Zador, P.L., 1982. Asymptotic quantization error of continuous signals and the quantization dimension. *IEEE Trans. Inform. Theory*, **28**(2):139-149. http://dx.doi.org/10.1109/TIT.1982.1056490