electronics

MDPI

Article

An Efficiency–Accuracy Balanced Power Leakage Evaluation Framework Utilizing Principal Component Analysis and Test Vector Leakage Assessment

Zhen Zheng *, Yingjian Yan, Yanjiang Liu, Linyuan Li and Yajing Chang

Cryptographic Institute, Information Engineering University, Zhengzhou 450001, China
* Correspondence: zhengzhen_0917@163.com

Abstract: The test vector leakage assessment (TVLA) is a widely used side-channel power leakage detection technology which requires evaluators to collect as many power traces as possible to ensure accuracy. However, as the total sample size of the power traces increases, the amount of redundant information will also increase, thus limiting the detection efficiency. To address this issue, we propose a principal component analysis (PCA)-TVLA-based leakage detection framework which realizes a more advanced balance of accuracy and efficiency. Before implementing TVLA to detect leakage, we project the original power data onto their most significant feature dimensions extracted by the PCA procedure and screen power traces according to the magnitude of their corresponding components in the variance of the projection vector. We verified the overall performance of the proposed framework by measuring the detection capability and efficiency with *t*-values and the required time, respectively. The results show that compared with similar existing schemes, under the best circumstances, the proposed framework decreases the *t*-value by 4.3% while saving time by 25.2% on the MCU platform and decreases the *t*-value by 2.4% while saving time by 38.0% on the FPGA platform.

Keywords: side-channel; power information leakage; test vector leakage assessment; principal component analysis

Citation: Zheng, Z.; Yan, Y.; Liu, Y.; Li, L.; Chang, Y. An Efficiency–Accuracy Balanced Power Leakage Evaluation Framework Utilizing Principal Component Analysis and Test Vector Leakage Assessment. *Electronics* **2022**, *11*, 4191. https://doi.org/10.3390/electronics11244191

Academic Editor: Floriano De Rango

Received: 9 October 2022
Accepted: 13 December 2022
Published: 15 December 2022

1. Introduction

Side-channel power analysis has been shown to be a serious threat to cryptographic devices and algorithms. Over the past two decades, scholars have proposed various power analysis attacks, including differential power analysis (DPA), correlation power analysis (CPA), simple power analysis (SPA), and so on. Hence, evaluating the resistance of cryptographic implementations against power analysis attacks has become an important issue in the field of information security [1]. The evaluating process aims to judge whether there is side-channel leakage within a limited time [2]. Previously, the evaluating process was executed utilizing all kinds of known side-channel analysis attacks, with a successful key recovery by any attack indicating that there is leakage. Nevertheless, to cover all the possible cases, a large range of attacking approaches as well as hypothetical power models should be inspected to evaluate the possibility of key recovery. This methodology is becoming more challenging as the types of known side-channel attacks are steadily increasing [3–5]. Trivially, this time-consuming procedure cannot be comprehensive even if a large number of attacking forms and power models are examined [6], hence it becomes desirable to formulate side-channel leakage detection into a relatively deterministic procedure without considering too many attacking forms.

The most widely used technology among existing formulated assessments is the test vector leakage assessment (TVLA) proposed in 2011 by Goodwill et al. [7]. The principle of TVLA is that when there is leakage, the mean in terms of power consumption generated by the cryptographic device when processing different data is different; when there is no leakage, the mean of power consumption data corresponding to different data is not

significantly different. Based on Welch's *t*-test, TVLA performs a pointwise comparison of two divided sets of power traces under the null hypothesis that no leakage exists [8]. In addition its theoretical simplicity, the main advantage of TVLA is its low complexity, that is, by comparing only two classes of power data, TVLA reduces the detection problem to a trivial calculation and comparison task. However, the TVLA procedure is generally regarded as a tradeoff between the low complexity and the risk of false negatives and false positives [9]. False negatives occur when there is leakage, but TVLA fails to discover this, and false positives occur when there is no leakage, but TVLA declares this by mistake. Scholars have carried out a large number of studies on false positives and false negatives.

Related work. Ding et al. proposed a "higher criticism" test (HC test)-based TVLA process to carry out the HC test on the *p*-values at each sampling point obtained by the TVLA process and to judge the leakage by comparing the difference between the expected distribution of the *p*-values under the no-leakage circumstance and the distribution of the obtained *p*-values [10]. When there are multiple leakage signals in the power trace, this process can detect the leakage with fewer power traces. A paired *t*-test process was proposed by Ding et al. to pair two adjacent encryptions of the cryptographic algorithm [11]. Since the time to perform one encryption is very short, the external environment of the two adjacent encryptions is roughly the same. Therefore, making the subtraction of the power consumption of the two paired encryptions will reduce the impact of environmental noise, thus detecting leakage with fewer power consumption data. In addition, Moradi et al. proposed using the combination of the chi-square test and TVLA to detect leakage [12]. This proposal can naturally capture the power leakage in multiple statistical moments. Bache et al. proposed a novel leakage assessment method based on the confidence interval [13] which does not increase the cost of the power consumption measurement and has strong robustness to noise.

Theoretically, TVLA confirms leakage by demonstrating a statistically nonnegligible difference between the means of two power consumption sets. However, when the sample size in terms of power traces is too small, the obtained difference may be accidentally too small or too large to coincide with that of the power consumption population, and thus one may ignore the existing leakage or confirm a leakage that does not actually exist. To address this issue, as many as possible power traces should be collected to exclude the influence of accidental factors such as noise and so on. Hence, many techniques controlling false negatives and false positives require an increase in the sample size in terms of power traces [14]. However, the power traces will be doped with too much redundant information when the sample size increases to a large number. Furthermore, all the sampling points in the power traces are detected one by one during the implementation of TVLA, and the redundant information will be applied to all the sampling points, multiplying the data processing workload and seriously restricting the detection efficiency.

In this work, we propose eliminating the redundant information in the original power consumption data by screening the power traces. The thought of screening power traces was first put forward by Tiri et al. [15], who found that the success rate of attack aiming at masked cryptographic implementation was improved by screening power traces. However, this methodology is only applicable to certain specific mask schemes and inevitably has a number of limitations. Kim et al. found that the power consumption at each sampling point in power traces follows a certain normal distribution and accordingly proposed screening power traces based on the normal distribution [16]. However, this normal distribution-based method has to locate the exact sampling points corresponding to the sensitive intermediate value of the algorithm in advance. Hu et al. improved this method, but the operation proved to remain difficult even after the modification [17]. Kim et al. proposed screening power traces by reducing the dimension of the power consumption data [18]. They mathematically analyzed the power data to select power traces based on dimension reduction, that is, they used a relatively low-dimensional vector to represent the characteristics of the original data while requiring the original information of the data to be preserved as much as possible. This scheme does not need to locate the corresponding

sampling points of the sensitive intermediate values, but the quality of the selected power traces fluctuates severely.

Our Contribution. In this work, we apply principal component analysis (PCA) [19,20] to extract the main feature dimensions of the power consumption data. Accordingly, we put forward a novel PCA-TVLA-based power information leakage detection framework, which selects power traces to facilitate the TVLA procedure. According to the verification results, our technique realizes a more advanced balance of detection accuracy and efficiency compared with the existing detection schemes.

Outline. The rest of the paper is organized as follows. Section 2 presents specific basic background knowledge, including a concise analysis of the principle of side-channel power leakage detection as well as the specific TVLA technology process. Section 3 introduces the PCA procedure and briefly analyzes its principle. Then, the power traces screening algorithm is proposed, based on which we construct a novel PCA-TVLA leakage detection framework. Experimental verification was carried out based on a Microcontroller Unit (MCU) and FPGA, and the results are presented in Section 4. Section 5 concludes the paper.

2. Preliminaries

2.1. Leakage Detection Principle

The power consumption of a cryptographic device depends on the cryptographic operation and the processed data, wherein the component dependent on the operation can be denoted as P_{op} and the component dependent on the processed data can be denoted as P_{data}. Simultaneously, the power consumption also contains a constant part, P_{const}, caused by transistor processing of the current detection, rectification, amplification, switching, voltage regulation, signal modulation, and other motions unrelated to the cryptographic operation and the processed data. Unavoidably, there is also the influence of noise, P_{noise}. Accordingly, the total power of the cryptographic device can be formulized by Equation (1) [21]:

$$P_{total} = P_{op} + P_{data} + P_{noise} + P_{const} \tag{1}$$

In the TVLA procedure, sampling points in the power trace are detected one by one. A certain sampling point in the power trace corresponds to a specific intermediate operation of the cryptographic algorithm, and thus the P_{op} part of each power consumption value at a certain sampling point is equal. Moreover, the constant part, P_{const}, is also equal at a certain sampling point by definition. Therefore, the fluctuation of different power values at the same sampling point is caused by P_{noise} and P_{data}. In power analysis attacks, attackers try to eliminate the influence caused by P_{noise} and to utilize the statistical characteristics of P_{data} to design various attacks to acquire information related to the processed data, thereby recovering the key. Therefore, if a certain statistical characteristic of the power consumption at one or multiple sampling points in the power trace change as the processed data change, it indicates that there is information available for the attacker at the corresponding sampling point, that is, there is power information leakage.

2.2. Leakage Detection Using TVLA

TVLA is essentially a hypothesis testing process based on Welch's t-test, and its null hypothesis and alternative hypothesis are $\mathbf{H}_0$: there is no leakage and $\mathbf{H}_1$: there is leakage, respectively. The objective of Welch's t-test is to provide a quantitative value as a probability that the $\mathbf{H}_0$ is true, i.e., the power consumption samples in the two divided sets were drawn from the same population. The main steps of TVLA are as follows: Let n_0 (resp. n_1), μ_0 (resp. μ_1), and S_0^2 (resp. S_1^2) stand for the cardinality, the sample mean, and the sample variance of the set L_0 (resp. L_1), respectively. The t-test statistic t-value and the degree of freedom v are computed as

$$t = \frac{\mu_0 - \mu_1}{\sqrt{\frac{{S_0}^2}{n_0} + \frac{{S_1}^2}{n_1}}}, v = \frac{\left(\frac{{S_0}^2}{n_0} + \frac{{S_1}^2}{n_1}\right)^2}{\frac{\left(\frac{{S_0}^2}{n_0}\right)^2}{n_0 - 1} + \frac{\left(\frac{{S_1}^2}{n_1}\right)^2}{n_1 - 1}} \tag{2}$$

In cases where $s_0 \approx s_1$ and $n_0 \approx n_1$, the degree of freedom can be estimated by $v \approx n_0 + n_1 = n$. As the final step, the probability to accept the null hypothesis is estimated by employing the Student's t distribution probability density function:

$$f(t, v) = \frac{\Gamma\left(\frac{v+1}{2}\right)}{\sqrt{\pi v}\Gamma\left(\frac{v}{2}\right)}\left(1 + \frac{t^2}{v}\right)^{-\frac{v+1}{2}} \tag{3}$$

where $\Gamma(.)$ denotes the gamma function. The desired probability is calculated as

$$p = 2\int_{|t|}^{\infty} f(t, v)dt \tag{4}$$

As an alternative, the corresponding cumulative distribution function can be used:

$$F(t, v) = \frac{1}{2} + t\Gamma\left(\frac{v+1}{2}\right)\frac{2^{F_1}(\frac{1}{2}, \frac{v+1}{2}, \frac{3}{2}, -\frac{x^2}{v})}{\sqrt{\pi v}\Gamma\left(\frac{v}{2}\right)} \tag{5}$$

where 2^{F_1} denotes the hypergeometric function. Hence, the result of the *t*-test can be obtained as

$$p = 2F(-|t|, v) \tag{6}$$

It can be easily deduced that small *p*-values (or, alternatively, large *t*-values) give explicit evidence to reject the null hypothesis and conclude that there is leakage. For the sake of simplicity, a threshold $|t| > 4.5$ is usually defined to reject the null hypothesis without considering the degree of freedom and the cumulative distribution function. This simplification is based on the inequality that $2 \times tcdf(t = -4.5, v > 1000) < 0.00001$, which leads to an accuracy of greater than 0.99999 to reject the null hypothesis [22].

To ensure the accuracy of results, the TVLA procedure requires evaluators to collect a large amount of power consumption data. A large amount of data provides a wealth of information, but it also includes a significant amount of redundant information and increases the workload of data analysis, which critically limits the detection efficiency. To alleviate the contradiction between detection efficiency and accuracy, this paper applies PCA to extract the main feature dimensions of the original power data, thereby filtering out the redundant information and realizing the screening of power traces.

3. Methodology

3.1. Theoretical Analysis of PCA Principle

PCA is widely used to reduce the number of predictive variables. In brief, PCA looks for a small number of linear combinations of the variables that can be used to summarize the data without losing too much information. For m pieces of n-dimension data, they can be denoted as a matrix $\boldsymbol{A}$ of size $m \times n$. PCA is the process of constructing new feature dimensions according to the mathematical characteristics of the original data and projecting the original n-dimension features to the newly established k-dimension features. The specific steps are as presented in Algorithm 1.

Algorithm 1 PCA Procedure

Input: the original matrix $\boldsymbol{A}$, number of feature dimensions k to retain;
Output: the projection matrix $\boldsymbol{E}$;
Steps:

1. Centralize the original matrix $\boldsymbol{A}$, i.e., subtract the mean of the corresponding column from each element in $\boldsymbol{A}$ to obtain matrix $\boldsymbol{B}$:

$$\boldsymbol{B} = centralize(\boldsymbol{A}) \tag{7}$$

2. Calculate the covariance matrix $\boldsymbol{C}_{n\times n} = \boldsymbol{B}^T\boldsymbol{B}$;
3. Perform the eigendecomposition on matrix $\boldsymbol{C}$ to obtain n eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ sorted from large to small and the corresponding eigenvectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n$;
4. Select the largest k eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_k$ and take the corresponding eigenvector $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k$ as the column vectors to form the matrix $\boldsymbol{D}_{n\times k}$;
5. Project:

$$\boldsymbol{E}_{\mathrm{m}\times\mathrm{k}} = \boldsymbol{B}_{m\times n} \times \boldsymbol{D}_{n\times k} \tag{8}$$

Through the above steps, the PCA procedure projects the original data into the newly established space with $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k$ as the coordinate axis. The eigenvector $\boldsymbol{v}_1$, corresponding to the largest eigenvalue, λ_1, coincides with the direction that maximizes the variance of the projection values of the original data, the eigenvector $\boldsymbol{v}_2$, corresponding to λ_2, is the direction that maximizes the variance of the projection values in the plane orthogonal to $\boldsymbol{v}_1$, and $\boldsymbol{v}_3$ is also the direction that maximizes the variance of the projection values in the plane orthogonal to both $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, and so on. The larger the variance of the projection of the original data in a certain direction, the more important the features defined in that direction are. PCA is inherently a feature extraction process that retains the main features of the original data and discards other unimportant features. Moreover, the magnitude of the eigenvalues indicates how important the corresponding features are. Under most circumstances, there is $\lambda_1 \gg \lambda_2 \gg \cdots \gg \lambda_n$ in the eigendecomposition, that is to say, the projection of the original data on $\boldsymbol{v}_1$ covers most of the important features of the original data. Therefore, the power traces can be sorted according to the projection values on $\boldsymbol{v}_1$ to select part of the traces that can represent the main characteristics of the original data, thereby filtering out excessive redundant information.

3.2. Power Traces Screening Algorithm Based on PCA

Referring to Equation (8), the original power consumption data can be projected to the eigenvector corresponding to the largest eigenvalue:

$$\boldsymbol{E}_{\mathrm{m}\times 1} = \boldsymbol{B}_{m\times n} \times \boldsymbol{D}_{n\times 1} = (centralize(\boldsymbol{A}))_{m\times n} \times \boldsymbol{D}_{n\times 1} \tag{9}$$

The projecting process in Equation (9) reveals that each element, $e_i (i \leq m)$, in $\boldsymbol{E}$ corresponds to each power trace in the original data matrix, $\boldsymbol{A}$. Thus, the power traces can be screened according to the statistical characteristics of the elements in $\boldsymbol{E}$. In the PCA procedure, $\boldsymbol{v}_1$ is the direction that maximizes the variance of the elements in the projection of the original data, and for this reason, $\boldsymbol{v}_1$ represents the most important feature direction of the original data. Similarly, the power traces can be sorted according to the corresponding components in the variance of $e_1, e_2, \ldots, e_m$. By definition, the variance of $e_1, e_2, \ldots, e_m$ is

$$\mathrm{var} = \frac{1}{m}\sum_{i=1}^{m}\left(e_i - \frac{1}{m}\sum_{j=1}^{m} e_j\right)^2 \tag{10}$$

where in the component corresponding to the element e_i is

$$\mathrm{var}_i = (e_i - \frac{1}{m}\sum_{j=1}^{m} e_j)^2 \tag{11}$$

Assuming that the projected vector is $\boldsymbol{E} = (e_1, e_2, e_3, e_4, e_5) = (1, 4, 5, 7, 8)$, the corresponding components of the elements in $\boldsymbol{E}$ can be trivially obtained as $(16, 1, 0, 4, 9)$ from Equation (11). Thus, the resulting sequence is e_1, e_5, e_4, e_2, e_3 successively. Finally, power traces can be screened according to their corresponding elements in this sequence. The above process is as presented in Algorithm 2:

Algorithm 2 Power traces screening algorithm based on PCA

Input: Original power data matrix $\boldsymbol{A}_{m\times n}$, number of power traces R to retain;
Output: Screened power data matrix $\boldsymbol{T}_{k\times n}$;
Steps:

1. Perform Algorithm 1 on the power data matrix $\boldsymbol{A}_{m\times n}$, wherein $k = 1$;
2. Calculate the corresponding variance component var_i of each element $e_i (1 \le i \le m)$ in $\boldsymbol{E}$;
3. Pick out the power traces corresponding to the R largest components to form matrix $\boldsymbol{T}$;
4. Output $\boldsymbol{T}$.

By screening the power traces, the amount of data processing in each execution of TVLA can be reduced, and thus the overall evaluation efficiency can be improved. Moreover, the methodology of feature extraction can retain the characteristics of the original power consumption data to a great extent [19,20], thus ensuring the accuracy of the leakage detection results. Based on the power traces screening algorithm, a leakage detection framework can be proposed.

3.3. PCA-TVLA-Based Leakage Detection Framework

The PCA-TVLA-based leakage detection framework can be obtained by combining Algorithm 2 with the TVLA procedure. The salient steps of the detection framework are shown in Figure 1.

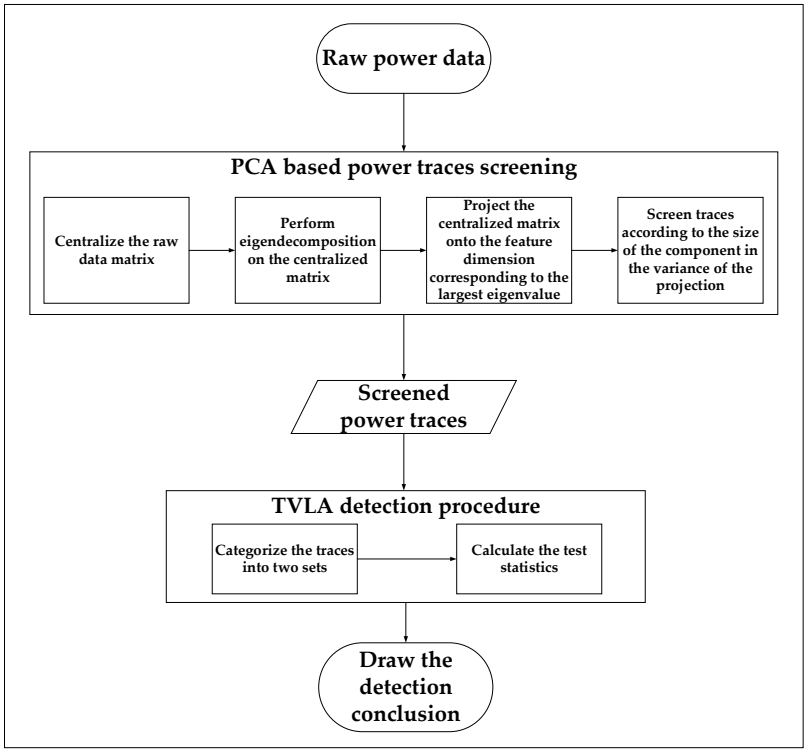

Figure 1. PCA-TVLA-based leakage detection framework.

The above framework aims at realizing an improvement in leakage evaluation efficiency under the premise of ensuring accuracy. As shown in Figure 1, Algorithm 2 is first applied to the original power data to filter out redundant information, thereby reducing the workload of the evaluation procedure. Simultaneously, benefiting from the properties of PCA, the screened power traces retain most of the information and characteristics of the original power data, thus guaranteeing detection accuracy. Subsequently, the TVLA procedure is performed on the screened power data to examine whether there is leakage.

4. Experimental Verification

The power consumption data in this section were collected from the Chipwhisperer MCU platform and DPA contest v2 FPGA-based datasets (http://www.dpacontest.org, accessed on 1 January 2022). The performance of the proposed framework was verified in terms of aspects of detection potency and detection efficiency, wherein the detection potency refers to the capability to detect leakage at a given power trace sample size, with t-values as the indicator, and the detection efficiency refers to the detection speed rate, with the required time as the indicator.

4.1. Verification on the MCU Platform

In this part, the power consumption of the AES-128 algorithm running on the Chipwhisperer development board was captured to implement the verification. The power-capturing device is shown in Figure 2.

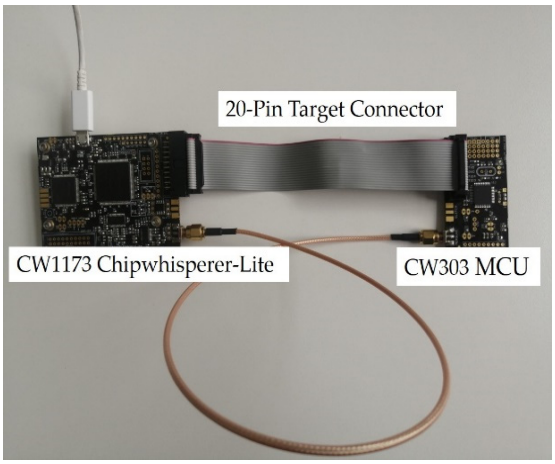

Figure 2. The connection of the power collection device.

As shown in Figure 2, the target board is the CW303 MCU, on which the AES-128 algorithm runs, and the capturing board is the CW1173 Chipwhisperer-Lite, which is not equipped with an oscilloscope and directly captures the power consumption through its OpenADC module. One of the power traces collected is shown in Figure 3.

The cryptographic operations of the AES-128 algorithm, including AddRoundKey, SubBytes, ShiftRows, MixColumns, and so on, run serially on the MCU platform. Thus, in Figure 3, different waveforms represent different cryptographic operations. TVLA is a pointwise detection procedure. We set 2650 sampling points on each power trace, and thus we obtained 2650 t-values in a certain test. We carried out our verification by comparing the following four schemes:

1. The original TVLA [7], hereinafter referred to as Scheme 1;
2. The proposed PCA-TVLA-based leakage detection framework, hereinafter referred to as Scheme 2;
3. The screening of the power traces using the normal distribution-based method [16], followed by the implementation of TVLA on the retained traces, hereinafter referred to as Scheme 3;

4. The screening of the power traces using the dimension reduction-based method [18], followed by the implementation of TVLA on the retained traces, hereinafter referred to as Scheme 4.

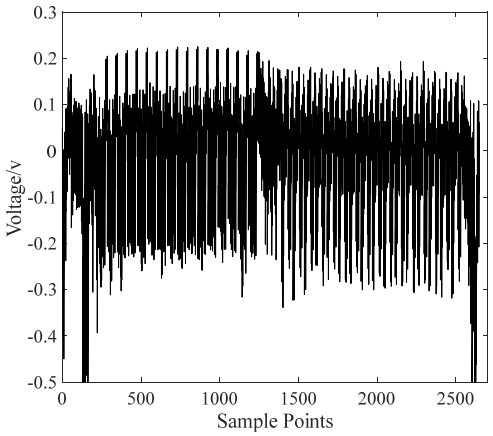

Figure 3. A piece of power trace collected from the Chipwhisperer MCU platform.

In the experimental procedure, 80% of the power traces were retained for leakage detection in Schemes 2, 3, and 4. To verify the accuracy of the evaluation, the *t*-values were recorded to examine the potency of leakage detection. Figure 4 presents the *t*-values obtained by each scheme at different sampling points under a total sample size (the sample size before screening power traces) of 500.

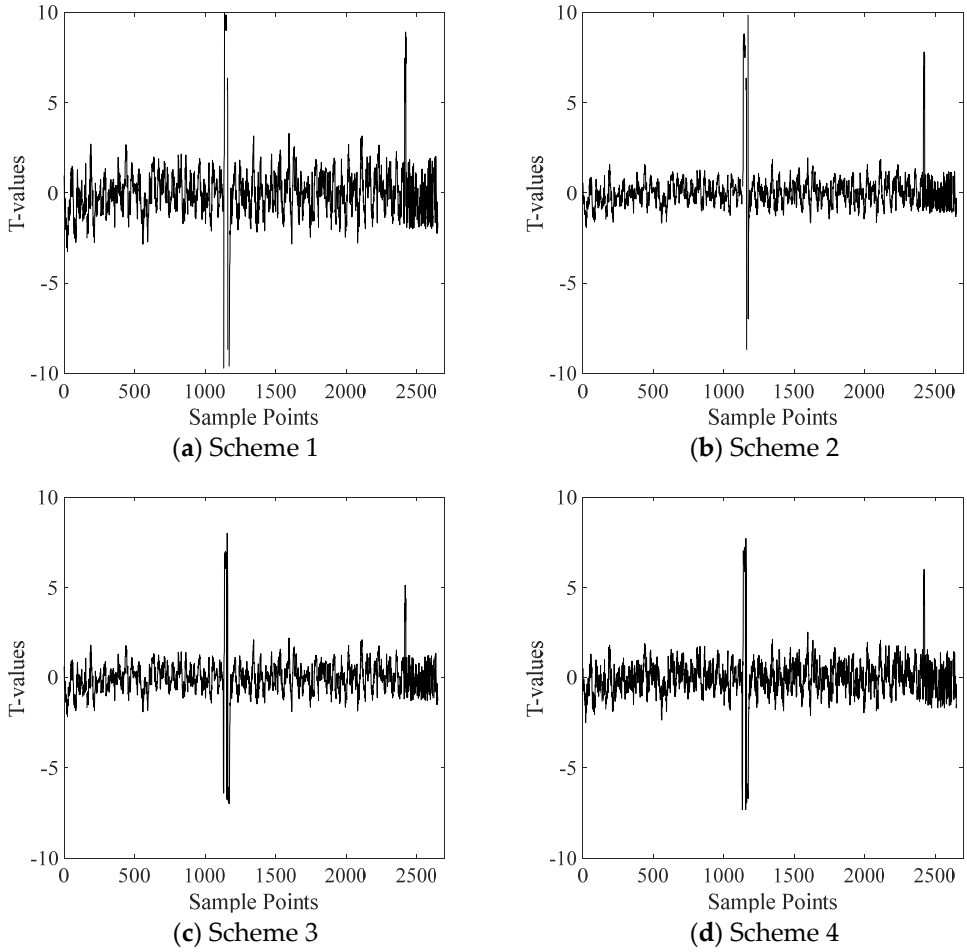

Figure 4. T-values of each scheme under a total sample size of 500.

The TVLA procedure confirms the leakage as long as any one sampling point of the traces rejects the null hypothesis. In other words, the device being tested is considered leaky when the maximum t-value is larger than the threshold (or, equivalently, when the minimum p-value is smaller than the threshold). That is, the TVLA is in fact a statistical maximum t-value (minimum p-value) test procedure [10]. As shown in Figure 4, the four schemes all obtained the maximum and the minimum t-values within the sampling point interval (1130, 1175), and most t-values within this interval exceeded the detection threshold of 4.5, indicating that there was power information leakage. Table 1 presents the maximum and the minimum t-values and the sequence numbers of the corresponding sampling points that obtained the t-values (referred as "The corresponding sampling point number" in Table 1) in the power traces.

Table 1. The statistics of the t-values under a total sample size of 500.

	Items	Scheme 1	Scheme 2	Scheme 3	Scheme 4
Maximum	The t-values	9.898	9.869	8.010	8.119
	The corresponding sampling point number	1138	1172	1157	1157
Minimum	The t-values	−9.873	−8.698	−7.382	−7.126
	The corresponding sampling point number	1133	1162	1173	1132

It can be observed from Table 1 that the maximum (minimum) t-values of the four schemes were not located in the same sampling point in the traces, but they were close to each other. This is because after preprocessing (applying PCA to screen the power traces), the retained power data of each method are different, but the locations of leakage in power traces are fixed. Additionally, Scheme 1 resulted in the largest t-value under a total sample size of 500; the maximum t-values obtained by the other three schemes are 9.869, 8.010, and 8.119, respectively, which are all smaller than the 9.898 of Scheme 1. The reason for this is that the power traces screening procedure inevitably filtered out useful information while eliminating redundant information. The minimum t-values were of the same circumstance. Compared with Scheme 1, Scheme 2 was 0.29% and 11.90% lower in terms of the maximum and minimum t-values, respectively, while Scheme 3 and Scheme 4 reached (19.07%, 25.23%) and (17.97%, 27.82%), respectively. Thus, we can speculate that the proposed Scheme 2 is more accurate than Schemes 3 and 4 under a total sample size of 500. To further verify the accuracy of the proposed framework, the above four schemes were implemented under different total sample sizes in terms of power traces and the obtained maximum t-values were recorded. The results are shown in Figure 5.

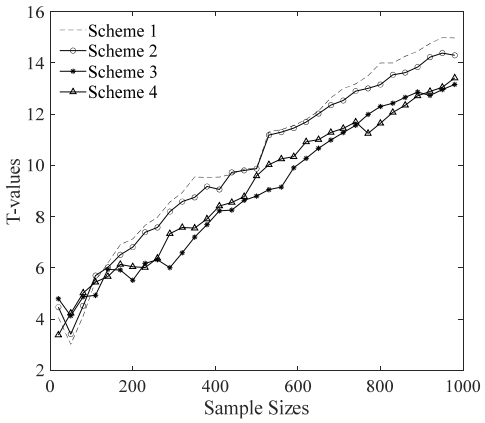

Figure 5. The maximum t-values of each scheme under different total sample sizes on the MCU platform.

As in Figure 5, the maximum *t*-values of the four schemes increased with the increase in the total power traces sample size, and the maximum *t*-values of Schemes 1 and 2 were higher than those of Schemes 3 and 4. When the total sample size was less than 200, the *t*-values obtained by the four schemes showed an alternating upward trend. When the total sample size came to more than 200, the *t*-values of Schemes 1 and 2 were similar, and both of them were larger than those of Schemes 3 and 4. When the total sample size was 900, the *t*-value of Scheme 2 was 4.3% smaller than that of Scheme 1 and was 7.5% and 8.8% larger than that of Scheme 3 and Scheme 4, respectively. Thus, it can be concluded that the detection potency of Scheme 1 is slightly higher than that of Scheme 2 and that both of them are higher than Schemes 3 and 4. Overall, the proposed detection framework guarantees adequate detection potency and accuracy.

To verify the leakage detection efficiency of the proposed framework, the performing times of each scheme under different total sample sizes were recorded. The results are depicted in Figure 6.

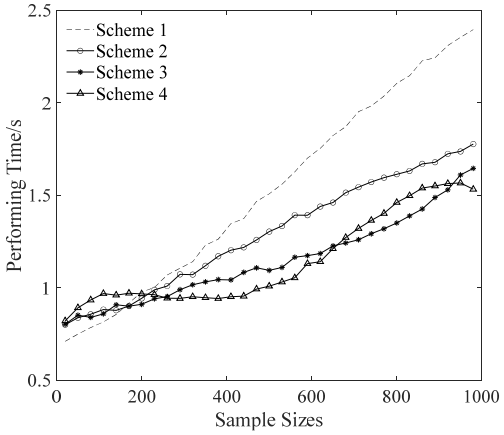

Figure 6. Performing time of each scheme under different total sample sizes on the MCU platform.

Table 2 presents the consumed time of the four schemes under different total sample sizes.

Table 2. The performing time on the Chipwhisperer MCU platform.

Total Sample Sizes	Scheme 1	Scheme 2	Scheme 3	Scheme 4
100	0.816 s	0.883 s	0.859 s	0.969 s
300	1.106 s	1.073 s	0.991 s	0.970 s
500	1.509 s	1.304 s	1.095 s	1.010 s
700	1.953 s	1.545 s	1.261 s	1.321 s
900	2.245 s	1.680 s	1.489 s	1.551 s

As illustrated in Figure 6 and Table 2, when the total sample size was less than 200, Scheme 1 consumed less time than the other three schemes, and the consumed times of the other three schemes were similar to each other. When the total sample size came to more than 200, Scheme 1 consumed more time than the other three schemes, with this owing to the fact that the total performing time consists of the preprocessing time and the detecting time, i.e., the evaluators apply PCA on the original power data to screen power traces in the preprocessing stage and implement TVLA on the retained power consumption data to determine whether there is leakage in the detecting stage. Although Scheme 1 required the shortest preprocessing time, the other three schemes screened power traces in the preprocessing stage, thus reducing the data processing amount in the leakage detecting stage and shortening the detecting time at each sampling point, so that the total performing

time was reduced when the total sample size was more than 200. When the total sample size was 900, the consumed time of Scheme 2 (the proposed framework) was 25.2% shorter than that of Scheme 1 and was 12.8% and 8.3% longer than that of Scheme 3 and Scheme 4, respectively. Therefore, the proposed Scheme 2 required less time than Scheme 1 and required a similar amount of performing time compared to Schemes 3 and 4.

To summarize the above results, compared with Scheme 1, the proposed Scheme 2 significantly improved the leakage detection efficiency while slightly reducing the detection potency and accuracy. Compared with Schemes 3 and 4, on the premise of significantly improving the detection potency and accuracy, the proposed Scheme 2 slightly reduced the detection efficiency. Overall, the performance of the proposed framework is superior to the existing methods.

4.2. Verification of The FPGA Power Data Set

The power data in this portion of the paper were collected from the highly recognized power datasets DPA contest v2, which captured the power consumption of the AES-128 algorithm on the FPGA development board SASEBO GII. One of the power traces is shown in Figure 7.

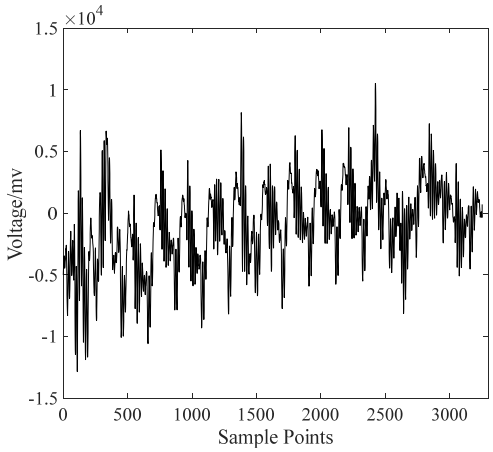

Figure 7. A piece of power trace collected from the FPGA platform.

On the FPGA platform, the cryptographic operations run in parallel. A certain sampling point may correspond to more than one cryptographic operation. Therefore, to detect the existing leakage, more power traces are required. There are 3253 sampling points in each trace. Thus, we obtained 3253 *t*-values during a specific detection. Referring to the experimental settings in Section 4.1, the maximum *t*-values of the aforementioned four schemes under different total sample sizes were recorded and the results are depicted in Figure 8.

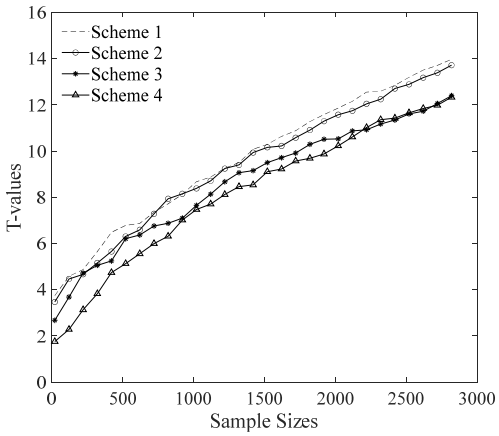

Figure 8. The maximum *t*-values of each scheme under different total sample sizes on FPGA.

As in Figure 8, the maximum *t*-values obtained by Schemes 1 and 2 were similar under different total sample sizes, and both of them were larger than those of Schemes 3 and 4. When the total sample size of power traces came to more than 1500, the gap between Schemes 1 and 2 and Schemes 3 and 4 increased with the increase in the total sample size. When the total sample size was 2500, the difference in *t*-values between Schemes 1 and 2 was 0.31 and the gap between Scheme 2 and Schemes 3 and 4 came to be 1.23 and 1.25, respectively, that is, the *t*-value of Scheme 2 was 2.4% smaller than that of Scheme 1 and was 10.6% and 10.7% higher than that of Schemes 3 and Scheme 4, respectively. Therefore, the detection potency of the proposed framework was roughly equivalent to that of Scheme 1 and was significantly higher than that of Schemes 3 and 4. Subsequently, the total performing times of each scheme under different total sample sizes were recorded and are shown in Figure 9.

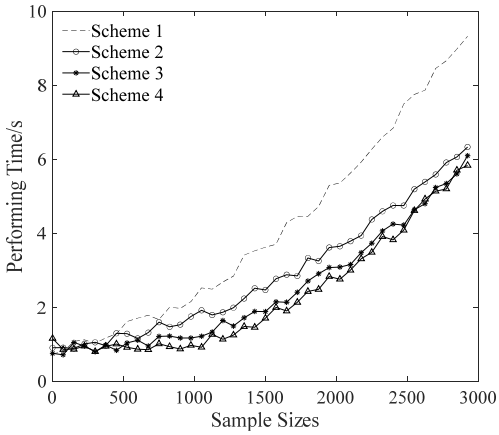

Figure 9. Total performing time of each scheme under different total sample sizes on the FPGA platform.

Table 3 presents the statistical results under total sample sizes of 500, 1000, 1500, 2000, and 2500.

Table 3. The performing times on the FPGA platform.

Total Sample Sizes	Scheme 1	Scheme 2	Scheme 3	Scheme 4
500	1.405 s	1.179 s	0.956 s	0.876 s
1000	2.247 s	1.791 s	1.259 s	1.131 s
1500	3.565 s	2.560 s	1.883 s	1.741 s
2000	4.981 s	3.632 s	3.013 s	2.870 s
2500	7.309 s	5.296 s	4.417 s	4.252 s

As can be seen from Figure 9 and Table 3, when the total sample size was less than 500, the total performing time of the four schemes was similar. This is because the total sample size was small, and thus the preprocessing time and the detecting time were both short. When the total sample size was more than 500, the consumed time of Scheme 1 began to soar, with it being much higher than that of the other three schemes because Schemes 2, 3, and 4 decreased the amount of data processing in the detecting phase by screening power traces during the preprocessing phase. When the total sample size was 2500, Scheme 2 consumed roughly 2 s less than Scheme 1 and roughly 1 s more than Schemes 3 and 4, i.e., the consumed time of Scheme 2 was 38.0% shorter than Scheme 1 and was 19.9% and 24.5% longer than Scheme 3 and Scheme 4, respectively. Moreover, in our experimental results, the proposed framework saved relatively little time due to the small sample size in terms of the power traces and the small sample size in terms of sampling points in the power traces. With the continuous optimization of side-channel protection technology and the

concurrent upgrading of side-information capturing equipment, it is increasingly necessary to analyze larger sample sizes in terms of power data. The proposed framework has strong guiding significance and can save more time in leakage situations with larger sample sizes.

To sum up, compared with Scheme 1, the efficiency of the proposed framework was greatly improved while ensuring detection potency and accuracy. Compared with Schemes 3 and 4, the proposed framework improved the detection potency and accuracy while slightly reducing the efficiency. That is, the performance of the proposed framework is more advanced than the existing methods on both MCU and FPGA platforms.

5. Conclusions

To ensure accuracy, the TVLA requires the capturing of as much power consumption data as possible, and, as a result, too much redundant information is adulterated in the collected data, significantly reducing the detection efficiency. In this work, a PCA-TVLA-based power leakage detection framework was proposed to address this issue. PCA was applied to the original power data matrix to extract the main features, and then certain power traces were screened out for TVLA leakage detection.

To verify the performance of the proposed scheme, experiments were carried out on the MCU and FPGA platforms, and the results showed that compared with the TVLA, the proposed scheme significantly improves the leakage detection efficiency on the premise of sufficient detection potency. Compared with other power traces screening methods, the proposed method slightly increased the total performing time and significantly improved the detection potency. Overall, the performance of the proposed method is better than the existing methods.

In the follow-up research, we will consider extending the proposed detection framework to the field of power analysis attacks, and we will continue to optimize TVLA and other power information leakage detection techniques.

Author Contributions: Conceptualization, Z.Z. and Y.Y.; methodology, Z.Z.; software, Y.C.; validation, Z.Z.; formal analysis, L.L.; investigation, Z.Z.; writing—original draft preparation, Z.Z.; writing—review and editing, Z.Z. and Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Randolph, M.; Diehl, W. Power side-channel attack analysis: A review of 20 years of study for the layman. *Cryptography* **2020**, *4*, 15. [CrossRef]
2. Bokharaie, S.; Jahanian, A. Side-channel leakage assessment metrics and methodologies at design cycle: A case study for a cryptosystem. *J. Inf. Secur. Appl.* **2020**, *54*, 102561. [CrossRef]
3. Liu, C.; Chakraborty, A.; Chawla, N.; Roggel, N. Frequency throttling side-channel attack. In Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Securitu, Los Angeles, CA, USA, 7–11 November 2022; pp. 1977–1991.
4. Wang, Y.; Paccagnella, R.; He, E.T.; Shacham, H.; Fletcher, C.W.; Kohlbrenner, D. Hertzbleed: Turning Power {Side-Channel} Attacks Into Remote Timing Attacks on x86. In Proceedings of the 31st USENIX Security Symposium (USENIX Security 22), Boston, MA, USA, 10–12 August 2022; pp. 679–697.
5. Ou, Y.; Li, L. Side-channel analysis attacks based on deep learning network. *Front. Comput. Sci.* **2022**, *16*, 1–11. [CrossRef]
6. Mather, L.; Oswald, E.; Bandenburg, J.; Wójcik, M. Does My Device Leak Information? An a priori Statistical Power Analysis of Leakage Detection Tests. In Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security, Singapore, 6–10 December 2021; Springer: Berlin/Heidelberg, Germany, 2013.
7. Gilbert Goodwill, B.J.; Jaffe, J.; Rohatgi, P. A testing methodology for side-channel resistance validation. In Proceedings of the NIST Non-Invasive Attack Testing Workshop, Nara, Japan, 26–27 September 2011; Volume 7, pp. 115–136.
8. Becker, G.; Cooper, J.; DeMulder, E.; Goodwill, G.; Jaffe, J.; Kenworthy, G.; Kouzminov, T.; Leiserson, A.; Marson, M.; Rohatgi, P.; et al. Test vector leakage assessment (TVLA) methodology in practice. In Proceedings of the International Cryptographic Module Conference, Gaithersburg, MD, USA, 24–26 September 2013; Volume 20.
9. Schnneider, T.; Moradi, A. Leakage assessment methodology—A clear roadmap for side-channel evaluations. *Cryptogr. Hardw. Embed. Syst. CHES* **2015**, *15*, 495–513.

10. Ding, A.A.; Zhang, L.; Durvaux, F.; Standaert, F.X.; Fei, Y. Towards sound and optimal leakage detection procedure. In Proceedings of the International Conference on Smart Card Research and Advanced Applications, Lugano, Switzerland, 13–15 November 2017; Springer: Cham, Switzerland, 2017; pp. 105–122.
11. Ding, A.A.; Chen, C.; Eisenbarth, T. Simpler, faster, and more robust t-test based leakage detection. In Proceedings of the International Workshop on Constructive Side-Channel Analysis and Secure Design, Graz, Austria, 14–15 April 2016; Springer: Cham, Switzerland, 2016; pp. 163–183.
12. Moradi, A.; Richter, B.; Schneider, T.; Standaert, F.X. Leakage detection with the x2-test. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2018**, *1*, 209–237. [CrossRef]
13. Bache, F.; Plump, C.; Güneysu, T. Confident leakage assessment—A side-channel evaluation framework based on confidence intervals. In Proceedings of the 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 19–23 March 2018; pp. 1117–1122.
14. Danial, J.; Das, D.; Ghosh, S.; Raychowdhury, A.; Sen, S. SCNIFFER: Low-cost, automated, efficient electromagnetic side-channel sniffing. *IEEE Access* **2020**, *8*, 173414–173427. [CrossRef]
15. Tiri, K.; Schaumont, P. Changing the odds against masked logic. In Proceedings of the International Workshop on Selected Areas in Cryptography, Montreal, QC, Canada, 14–18 August 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 134–146.
16. Kim, Y.; Sugawara, T.; Homma, N.; Aoki, T.; Satoh, A. Biasing power traces to improve correlation power analysis attacks. In Proceedings of the First International Workshop on Constructive Side-Channel Analysis and Secure Design (cosade 2010), Paris, France, 7–8 March 2010; pp. 77–80.
17. Hu, W.; Wu, L.; Wang, A.; Xie, X.; Zhu, Z.; Luo, S. Adaptive chosen-plaintext correlation power analysis. In Proceedings of the 2014 Tenth International Conference on Computational Intelligence and Security, Kunming, China, 15–16 November 2014; pp. 494–498.
18. Kim, Y.; Ko, H. Using principal component analysis for practical biasing of power traces to improve power analysis attacks. In Proceedings of the International Conference on Information Security and Cryptology, Seoul, Republic of Korea, 27–29 November 2013; Springer: Cham, Switzerland, 2013; pp. 109–120.
19. Abdi, H.; Williams, L.J. Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.* **2010**, *2*, 433–459. [CrossRef]
20. Hasan BM, S.; Abdulazeez, A.M. A review of principal component analysis algorithm for dimensionality reduction. *J. Soft Comput. Data Min.* **2021**, *2*, 20–30.
21. Mangard, S.; Oswald, E.; Popp, T. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*; Springer Science & Business Media: Berlin, Germany, 2008.
22. Durvaux, F.; Standaert, X. From improved leakage detection to the detection of points of interests in leakage traces. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, 8–12 May 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 240–262.