Journal of Electronic Testing (2023) 39:621–629
https://doi.org/10.1007/s10836-023-06090-7

Detection Method of Hardware Trojan Based on Attention Mechanism and Residual-Dense-Block under the Markov Transition Field

Shouhong Chen[2] · Tao Wang[2] · Zhentao Huang[2] · Xingna Hou[1,2]

Received: 27 July 2023 / Accepted: 19 October 2023 / Published online: 7 December 2023

Abstract
Since 2007, methods that utilize side-channel data to detect hardware Trojan (HT) problems have been widely studied. Machine learning methods are widely used for hardware Trojan detection, but with the development of integrated circuits (ICs), better results are usually obtained using deep learning methods. In this paper, we propose an architecture inspired by Residual-Block and Dense-Block and combine it with SE Attention Mechanism, which we named the Res-Dense-SE-Net network. By combining residual connectivity, dense connectivity, and attention mechanism, the Res-Dense-SE-Net network can enjoy the advantages of these three network architectures at the same time, which can improve the expressiveness and performance of the model. The Res-Dense-SE-Net network can capture the key features in the image better, and it can solve the problems of gradient vanishing and feature transfer efficiently, which can in turn improve the classification accuracy and the generalization ability of the model. Based on the publicly available AES series of hardware Trojans from TrustHub and the publicly available hardware Trojan-side channel data by Faezi et al., we evaluate the effectiveness of the method proposed in this paper. The experimental results show that when a single Trojan exists, the method proposed in this paper has a high accuracy rate; and when multiple types of hardware Trojans exist at the same time and need to be categorized, the categories of hardware Trojans can also be effectively identified, and the categorization accuracy is high compared with the existing deep learning methods.

Keywords Residual-Block · Dense-Block · SE-Block · MTF·Hardware · Trojan detection

1 Introduction

Since the late 1950s, integrated circuits have followed Moore's Law and have been rapidly evolving to become the main form of implementation of electronic products in everyday life. However, with the globalization of IC design, manufacturing, and sales, security issues are becoming more and more prominent, and hardware Trojans are one of the main threats. The research and development, mass production, and final deployment of a chip often require the cooperation of dozens of teams. However, in the implementation process, there may be attackers hidden in every team involved, who can maliciously tamper with the original design to achieve ulterior motives. This type of malicious tampering with the original circuitry in the form of hardware is known as a "hardware Trojan".

The real emergence of hardware Trojans as a specialized academic term in security began in 2007 [1]. With the advances in processes, a large-scale digital integrated circuit typically contains hundreds of millions of transistors, whereas a hardware Trojan typically contains only a few hundred logic gates at most. In other words, hardware Trojan circuits are very small in size relative to their parent circuits, and their physical-electrical properties are so weakly expressed that they are difficult to detect. In addition, hardware Trojan circuits are triggered only in very few cases where the conditions preset by the attacker are met. As a result, hardware Trojans are extremely stealthy. Meanwhile, hardware Trojans can cause serious damage such as denial of service, unexpected failures, data leakage,

Responsible Editor: C. A. Papachristou

✉ Xingna Hou
hxngl@guet.edu.cn

[1] School of Architecture and Transportation Engineering, Guilin University of Electronic Technology, Guilin 541000, China

[2] School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin 541000, China

and performance degradation of chips. A hardware Trojan structural module generalized by a simple abstraction can be divided into a trigger circuit responsible for activating the Trojan and a load circuit that determines the effect of the Trojan attack [15].

Since the activation mechanism and attack load of hardware Trojans are uncertain to the detector, the detector cannot utilize automated test vector generation tools to generate tit-for-tat test vectors for hardware Trojans in the same way that it is possible to test for hardware failures. The side channel analysis detection method for hardware Trojans essentially transforms the detection of hardware Trojans into a data classification problem by building a mathematical model based on its physical and operational parameters from its working mechanism. The accuracy of the model depends heavily on the size of the effective parameter inputs to the model and the ability to fit the function.

The validation experiments of the method proposed in this paper are based on the power and EM side-channel signaling datasets of the hardware Trojan benchmark. The contributions of this paper are as follows: 1. preprocessing the original data by using the Markov Transition Field(MTF), which highlights the temporal order between the data and makes the features more obvious; 2. designing a new network structure model for hardware Trojan detection, which improves the Residual-Block and Dense-Block structures based on the two, and uses a combination of the two to the new Residual-Dense-block is formed, and the SE structure is also inserted into the module, finally completing the Res-Dense-SE-Net network. Using this network structure to detect hardware Trojans, we can effectively improve the accuracy rate and achieve excellent recall, F1 score, and precision values; 3. The network proposed in this paper also has a very high accuracy rate in multi-Trojan classification, and can efficiently determine and differentiate between the types of hardware Trojans; 4. We surveyed existing HT detection methods and their shortcomings and depicted the current status and challenges of the research field, as well as the potential of Convolutional Learning for the potential for hardware security.

This paper is organized as follows: the second part is the background, the third part describes the analysis of the hardware Trojan detection method and algorithmic process proposed in this paper, the fourth part is the experimental results, and the fifth part is the conclusion.

2 Background

Side-channel analysis detection methods are the most diverse and widely researched methods in current hardware Trojan detection. Theoretically, a hardware Trojan exists in the form of a physical entity in the chip, and therefore its existence necessarily alters the non-logical nature of the original design, regardless of its ultimate purpose. Therefore, the core idea of the side-channel analysis detection method is to extract the side-channel information of the chip under test from the process deviation and measurement noise and compare it with the side-channel information of the ideal case, if the two are inconsistent, it means that the chip under test has been implanted with a hardware Trojan.

D. Agrawal et al. [1] creatively utilized power consumption information in side channel data to detect hardware Trojans in 2007. Since then, various side channel information has been attempted for detection [8, 15, 33]. For example, current regionalization analysis and insertion of correction circuits are used to improve the signal-to-noise ratio of dynamic power consumption [25, 37]. Salmani et al. achieved localized management of dynamic power consumption by reconfiguring the scan chain order, thus reducing the total dynamic power consumption of the parent circuit during Trojan detection [29]. In addition, S. Ghosh et al. proposed a multiparameter detection method that utilizes the correlation between two parameters, the maximum operating frequency, and the transient current, to compensate for the low detection sensitivity of the single-side channel method [9]. With the emergence of IP-level and bus-level hardware trojans and the increase in circuit size, researchers can continuously monitor chip runtime characteristics, including behavior, power consumption, etc. [4, 16]. The application of machine learning can be traced back to Jin et al. [17], who proposed in 2012 to use ANN models to process parameters measured from wireless encryption chips and achieve the classification of hardware Trojan embedded circuits. With the popularity of machine learning in 2014, this technology was widely applied in hardware Trojan detection [11, 23, 24, 27]. However, since machine learning cannot bring high accuracy, coupled with the rise of deep learning, many researchers choose to use deep learning for detection. In recent years, a large number of network structures based on CNN and RNN have been used in experiments, and faster detection speed and higher detection accuracy have proven the feasibility of deep learning in this direction [5, 22, 30, 32, 40, 41]. However, a single network structure cannot continuously improve the detection accuracy of related trojans. Continuously digging deep into the network not only greatly reduces the running speed of the program, but also easily leads to overfitting and reduces accuracy. Therefore, we came up with the method of using composite networks. In our investigation, we found that the attention mechanism can improve the accuracy of ResNet and Densenet [7, 13, 20, 21, 36, 38, 39], so we thought of combining ResNet and Densenet, inheriting the residual connection and dense connection, and introducing the SE attention mechanism to enhance the expression ability and importance of features. To make full use of the respective advantages of ResNet and Densenet, but also to make up for some of their limitations.

Up to now, many papers have proved that MTF, as a feature extraction and representation method in the process of time series processing, can play its advantages in many fields and improve the final classification performance. Therefore, on the premise of using MTF, this paper selects the composite network based on the combined structure of the Residual-Block and Dense-Block with an attention mechanism, to achieve the purpose of improving the network capacity and not excessive occupation of equipment computing resources.

3 Detection Model and Algorithmic Process

3.1 Base Structure

3.1.1 MTF

Markov transition field is a time series image coding method based on the Markov transition matrix. This method regards the time-lapse of time series as a Markov process, that is, when the current state is known, its future evolution does not depend on its past evolution, so it constructs a Markov transition matrix, and then expands it to a Markov transition field to realize image coding.

MTF was originally a feature extraction method for time series data. For time series data, MTF can capture the dynamic changes and conversion rules between time series. It can extract important features from data, especially for periodic or regular sequence data. MTF can express these features well, to better understand and analyze data. MTF transforms the original sequence data into a low-dimensional feature vector, which realizes the dimensionality reduction and compression of the data. This reduces data storage and computing costs and reduces redundant information while retaining critical information. By converting sequence data into MTF feature vectors, various machine learning algorithms can be used to realize data classification and recognition. The sequence data is converted into image form, to realize the visualization of data. This makes the data analysis and understanding more intuitive and easy to understand.

3.1.2 Residual-Block

A milestone in CNN's history is the emergence of the Resnet model. The core of the Residual-Block is to establish a "short circuit connection" between the front layer and the back layer, which helps to alleviate the problem of gradient disappearance and makes it easier to learn the expected mapping, to train a deeper network. Figure 1 shows a simplified version of the residual structure.

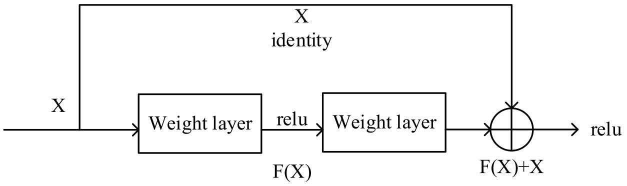

Fig. 1 Residual-Block

3.1.3 Dense-Block

The feature maps of each layer of a Dense-Block are the same size and can be connected in the channel dimension. The non-linear combination function in Dense-Block usually adopts the structure of BN + ReLU + 3 × 3 Conv. For the input of the later layer will be very large, a bottleneck layer will be used inside the Dense-Block to reduce the amount of calculation, mainly by adding 1 × 1 Conv to the original structure, as shown in Fig. 2.

3.1.4 Attention Mechanism

The se module assigns different weights to different positions of the image from the perspective of the channel domain through a weight matrix to obtain more important feature information. Mainly relying on squeeze and excitation, a 1*1*C weight matrix is obtained through a series of operations, and the original feature is reconstructed. The process is shown in Fig. 3 below (the number represents different channels, which is used to measure the importance of channels).

3.2 Experimental Preparation

All the experiments involved in this paper are completed in the cloud server. This environment is configured with 15 vCPU Intel(R) Xeon(R) platinum 8358P CPU, 80GB memory and RTXA5000(24GB), PyTorch 1.11.0/python3.8 (cuda11.3) as the operating environment, and the operating software version is Pycharm professional 2022.2.2.

The data source used in this paper is the public data set of IEEE Dataport [26], which is from the paper [5, 6]. The data involved in this data set are the power and EM side-channel signals of some HT references from TrustHub [28,

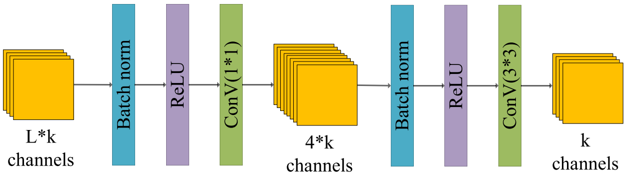

Fig. 2 Dense-Block with a bottleneck

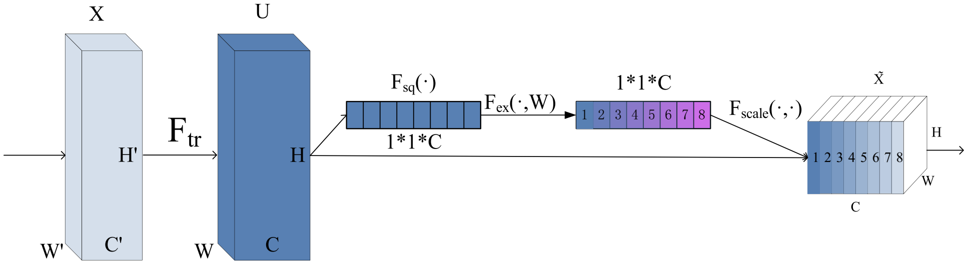

Fig. 3 The process of SE

31, 35]. All HTs are for an encrypted core circuit, which is named AES 128bits. AES circuit receives a 128-bit input value (plaintext), encrypts it with a key, and generates a 128-bit output (ciphertext). In each data collection experiment, the encryption process needs to be repeated 10,000 times to generate 10,000-time series signals.

The data selected in this paper uses two Sakura-G boards to randomly collect power data to consider process changes and reduce the impact. At the same time, the side channel data is collected in the presence of HT in two cases, namely, when HT is inactive and when HT is triggered. In this case, all measured power consumption data include the static power consumption of HT in addition to the power consumption of the underlying circuit. The only difference between the two data acquisition cases is the dynamic power consumption of HT. For possible noise interference, we believe that the network structure has a certain anti-interference ability to noise when independently training for each hardware Trojan, and can reduce the possible data impact caused by noise as much as possible. When the data is converted into waveform samples, it can be found that HT-inactive samplings have a similar set of features, while HT-triggered samplings have several different features from HT-inactive samplings. Therefore, we believe that these changes can be used to report HT-triggered cases using an exception detection mechanism. We use AES-T700 as an example. We only perform waveform conversion on the initial data, and we can get the example shown in the Fig. 4 below, where (a) (b) is the trigger state and (c) (d) is the inactive state. Paying attention to the images of the examples, we can see that the HT inactive samples all have a similar set of features, while the HT triggered samples have some features that are different from the HT inactive samples.

In this paper, to facilitate the visualization and comparison of experimental results, the Trojan horse types listed below are AES-T500, AES-T600, AES-T700, AES-T800, and AES-T1600. The time series signals of each HT are randomly extracted from the time series signals in the data set, of which 80% are used as the training set and 20% as the verification set.

We pre-processed the CSV format time serial side channel original data through MTF, changed it from low-dimensional to high -dimensional 224*224 RGB pictures, and sent it into the network structure for learning.

3.3 Algorithm Classification Process

In this experiment, the parameters selected by the network are verified by the control variable method. The parameters used to obtain the final results in the program are either the best choice under the same conditions, or a better solution under certain conditions (for example, after the adjustment, the running time is significantly reduced, but the accuracy rate is slightly reduced, so the parameters with slightly lower accuracy rate are selected).

This paper chooses the structure formed by the combination of residual block and dense block and attention mechanism as the backbone network of hardware Trojan horse detection. The schematic diagram of the network structure

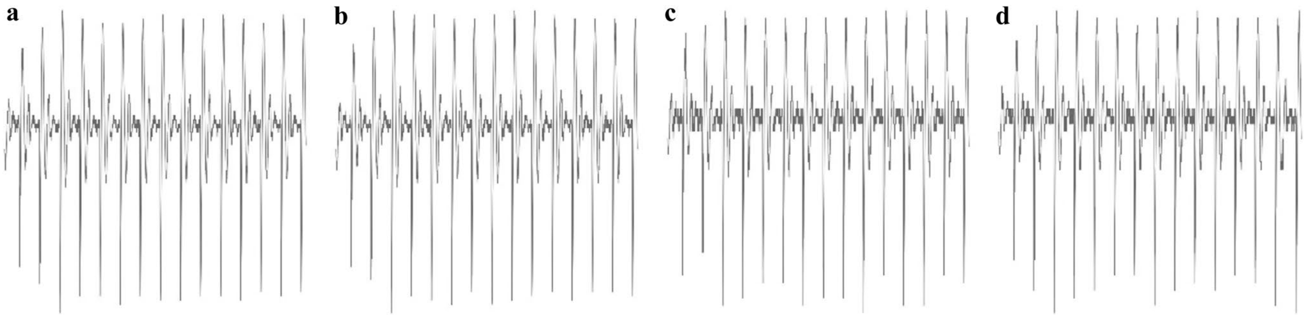

Fig. 4 Waveforms of AES-T700 (**a**, **b**, **c** and **d** from left to right)

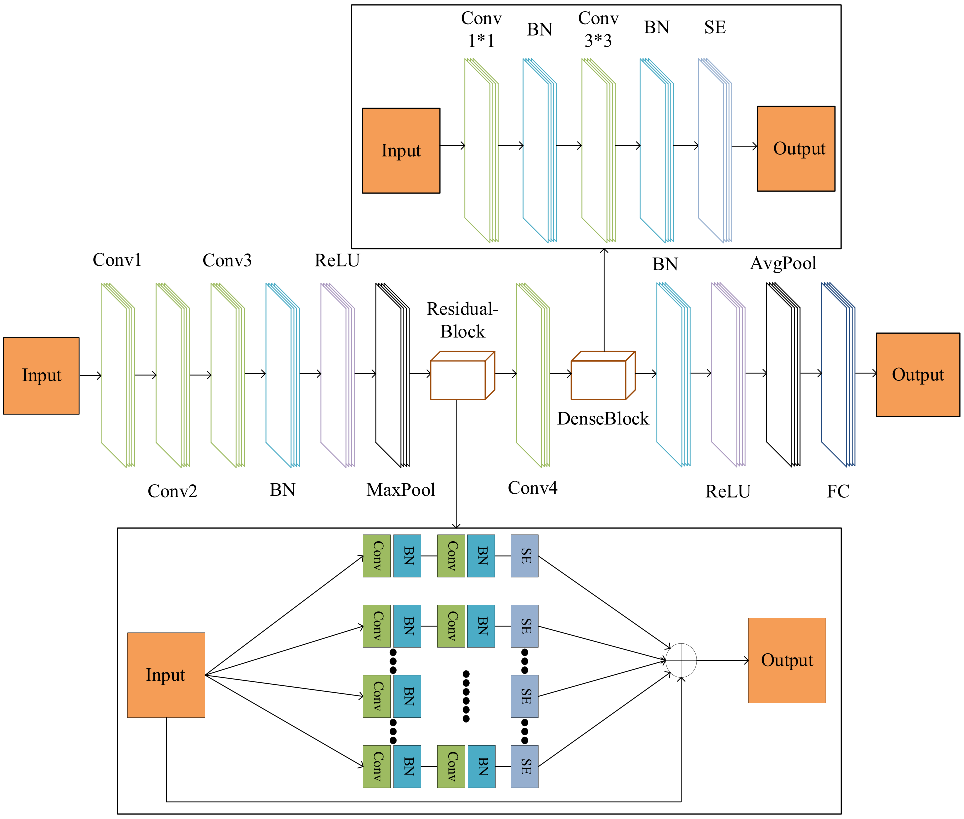

Fig. 5 Network structure

and the key parameters of the network are shown in Fig. 5 and Table 1 below.

Firstly, this paper uses MTF to preprocess the obtained time series signals, which can effectively ensure the extraction of the importance of data between the sequence of time, and also highlight the important features, to improve the accuracy of classification. Then, the converted RGB image is passed through the Res-Dense-SE-Net network. In the first half of the network, the Residual-Block is used for feature extraction, and the input image is connected to the depth residual after getting the feature map through the initial convolution layer and pooling layer of the Residual-Block. After a certain number of residual blocks, the output feature map of the Residual-Block is obtained. The output of Residual-Block is further processed in the second half using Dense-Block, where the output feature maps are fed into both the initial convolutional and pooling layers of Dense-Block to obtain the initial feature maps of Dense-Block. The initial feature map is passed into a dense block for a richer feature representation. Finally, the output feature maps are classified by a fully connected layer. And, we add an SE attention mechanism to Residual-Block and Dense-Block to enhance the expressiveness of the network. During the forward propagation of the model, the SE attention module is applied to the output of each residual and dense block to reinforce the key information in the feature map.

The input 224 * 224 RGB image is first subjected to 3 × 3 convolutions to extract shallow feature maps. Then, following the maximum pooling layer, the output size of the feature map is reduced while preserving its main features, thereby

Table 1 Key parameters of the network

	Layer	Parameter
	Conv1	kernel_size = (3, 3), stride = (2, 2), padding = (3, 3)
	Conv2	kernel_size = (3, 3), stride = (2, 2), padding = (3, 3)
	Conv3	kernel_size = (3, 3), stride = (2, 2), padding = (3, 3)
	Maxpool	kernel_size = 3, stride = 2, padding = 1
ResidualBlock	Conv	kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)
	Conv4	kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)
	Conv1*1	kernel_size = (1, 1), stride = (1, 1), padding = (1, 1)
DenseBlock	Conv3*3	kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)

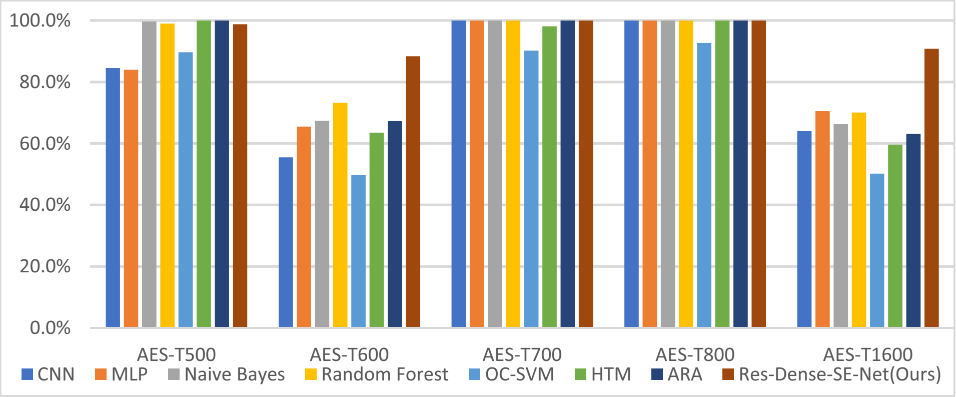

Fig. 6 Comparison of accuracy

reducing computational complexity. Then the feature map is sent to the Residual-Block and passed through a residual block format of [8, 8, 13, 33]. To enhance the network's focus on useful features for tasks, more effectively capture and utilize key information, and improve the model's performance and generalized ability, we have included an SE structure. Finally, it is input into the Dense-Block with the same SE structure. The attention mechanism added to it enhances the modeling ability of the model, enabling the network to more accurately understand and represent the structural information in the input data. After the series of operations, the network extracts deep abstraction and high-level semantic features, and ultimately passes through the global average pooling layer and fully connected layer.

This combined approach capitalizes on the respective advantages of Residual-Block and Dense-Block. Residual connections help to solve the gradient vanishing problem by allowing information to propagate faster through the network. Connecting through residual connections enables better training of deep networks and helps mitigate the effect of gradient vanishing on network performance. Dense connections allow features to fully propagate through the network, thereby improving the efficiency of feature reuse and information mobility, thus facilitating feature transfer and reuse, as well as better capturing features at different levels, and helping to mitigate the problem of information loss during feature transfer. The added SE attention mechanism can help the network to better focus on key features, enhance the expression of important features, and improve classification performance. By combining the structures, the advantages of these three network structures can be enjoyed simultaneously, which can improve the expressive ability and classification performance of the model. t can better capture the key features in the image and effectively solve the gradient vanishing and feature transfer problems, which in turn improves the classification accuracy and the generalization ability of the model.

4 Results

Due to the lack of open-source HT detection project models, replication of state-of-the-art HT detection methods on the side channel data used is not feasible. To make a quantitative comparison and further demonstrate the accuracy of the method proposed in this paper, we compare our detection mechanism with highly accurate models from [2, 3, 5, 6, 19]. The accuracy of the specific method is shown in Fig. 6 below.

From the above figure, it can be seen that the Res-Dense-SE-Net method proposed in this paper has high detection accuracy for several hardware Trojans activated, which is comparable to the state-of-the-art classification methods. This can reflect the feasibility of the methodology of this paper.

In addition, to be more intuitive, this paper also adopts the evaluation metrics commonly used in machine learning and deep learning to evaluate the classification performance of

Table 2 Other evaluation indexes for Single hardware Trojan type

	Recall/TPR	Precision	F1-score	TNR
AES-T500	0.986	0.990	0.988	0.976
AES-T600	0.898	0.870	0.884	0.870
AES-T700	1.000	1.000	1.000	1.000
AES-T800	1.000	1.000	1.000	1.000
AES-T1600	0.917	0.940	0.928	0.897

Table 3 data from other papers

	Average TPR	Average TNR	Average Accuracy
SVM	0.83	0.49	0.51
NN [12]	0.81	0.69	0.69
Multi-NN	0.85	0.70	0.73
NN [14]	0.72	0.90	/

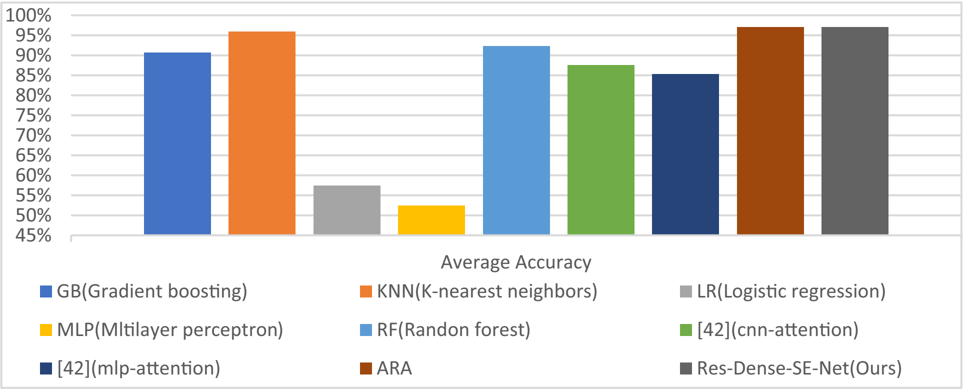

Fig. 7 Accuracy rate for mixed hardware Trojan types

the model, i.e., recall, precision, and F1 score. Their calculation formulas are as follows.

$$\mathrm{Re}\ call = \frac{TP}{TP + FN} \tag{1}$$

$$\mathrm{Pr}\ ecision = \frac{TP}{TP + FP} \tag{2}$$

$$F1 - score = \frac{2 * \mathrm{Pr}\ ecission * \mathrm{Re}\ call}{\mathrm{Pr}\ ecission + \mathrm{Re}\ call} \tag{3}$$

Of these, the triggered category of hardware Trojans is labeled "positive" and the inactive category is labeled "negative". Then we get the four categories of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). Based on the above definitions, we can also obtain the True Positive Rate (TPR) and True Negative Rate (TNR).

$$TPR = \mathrm{Re}\ call = \frac{TP}{TP + FN} \tag{4}$$

$$TNR = \frac{TN}{TN + FP} \tag{5}$$

Table 2 lists the assessment indicators of the proposal methods used to detect various hardware Trojans and the best results for the corresponding test set. Table 3 lists some metrics from the literature [10–12, 14], where "NN" refers to neural network-based approaches and "Multi-NN" refers to multi-intermediate layer neural networks. From the comparison of the data, it can be seen that the results of the method used in this paper are feasible when compared to the existing methods.

In addition, our proposed neural network is also effective in recognizing individual hardware Trojan types when multiple types of hardware Trojans coexist and need to be classified. The average detection accuracy is 97%, with the lowest accuracy for accurate identification of a single Trojan species being 93% and the highest being 100%. The results are also compared with existing general machine learning methods [18] and with [34] that have a similar case of multiple hardware Trojan types as this paper, and the results are shown in Fig. 7 below.

5 Conclusion

As the importance of hardware Trojan detection grows, new technology tools are constantly being used to detect potential hardware Trojans. From the initial machine learning to today's deep learning, the detection rate and detection accuracy of hardware Trojans have improved dramatically, and the false detection rate is decreasing as the network goes deeper and the level of equipment improves.

In this paper, we propose a combinatorial deep neural network "Res-Dense-SE-Net" based on a Residual-Dense combinatorial structure and attention mechanism under MTF preprocessing. This neural network enables the detection of hardware Trojans based on time series of real channel data without a golden chip. The method proves feasibility in terms of detection accuracy of hardware Trojans compared to existing detection methods. In addition, when multiple hardware Trojans exist in a mixture and need to recognize the types, the neural network can effectively recognize the types of Trojans among them, effectively differentiate between multiple hardware Trojans, and achieve high accuracy in accurately recognizing the types of hardware Trojans.

Funding This work was supported by the National Natural Science Foundation of China (No.61661013), Guangxi Natural Science Foundation (No. 2018GXNSFBA281175), the Basic Ability Enhancement Program for Young and Middle-aged Teachers of Guangxi (No.2022KY0197), Guangxi Key Laboratory of Automation Test and Instrumentation (No.YQ22102), Innovation Project of GUET Graduate Education (No. 2022YCXS162) and Innovation Project of GUET Graduate Education (No. 2023YCXS119).

Availability of Data and Materials The initial data that support the findings of this study are openly available in IEEE-DataPort at https://ieee-dataport.org, reference number [26].

Declarations

Conflicts of Interests All the authors declare no conflict of interest.

References

1. Agrawal D, Baktir S, Karakoyunlu D, Rohatgi P, Sunar B (2007) Trojan Detection using IC Fingerprinting. Proc.2007 IEEE Symposium on Security and Privacy (SP '07). Berkeley, CA, USA, pp 296–310
2. Bao C, Forte D, Srivastava A (2014) On application of one-class SVM to reverse engineering-based hardware Trojan detection. Proc. Fifteenth International Symposium on Quality Electronic Design. Santa Clara, CA, USA, pp 47–54
3. Chen S, Wang T, Huang Z, Hou X (2023) Detection method of Golden Chip-Free Hardware Trojan based on the combination of ResNeXt structure and attention mechanism. Comput Secur 134:103428
4. Dubeuf J, Hély D, Karri R (2013) Run-time detection of hardware Trojans: The processor protection unit. Proc.2013 18th IEEE European Test Symposium (ETS). Avignon, France, pp 1–6
5. Faezi S, Yasaei R, Al Faruque MA (2021) HTnet: Transfer learning for golden chip-free hardware trojan detection. Proc.2021 Design, Automation & Test in Europe Conference & Exhibition (DATE). Grenoble, France, pp 1484–1489
6. Faezi S, Yasaei R, Barua A, Faruque MAA (2021) Brain-inspired golden chip free hardware trojan detection. IEEE Trans Inf Forensics Secur 16:2697–2708
7. Farag MM, Fouad M, Abdel-Hamid AT (2022) Automatic severity classification of diabetic retinopathy based on densenet and convolutional block attention module. IEEE Access 10:38299–38308
8. Forte D, Bao C, Srivastava A (2013) Temperature tracking: an innovative run-time approach for hardware Trojan detection. Proc.2013 IEEE/ACM Int Conf Comput Aided Des (ICCAD). San Jose, CA, USA, pp 532–539
9. Ghosh S, Basak A, Bhunia S (2015) How secure are printed circuit boards against trojan attacks? IEEE Design & Test 32(2):7–16
10. Hasegawa K, Shi Y, Togawa N (2018) Hardware Trojan Detection Utilizing Machine Learning Approaches. Proc.2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). New York, NY, USA, pp 1891–1896
11. Hasegawa K, Yanagisawa M, Togawa N (2017) A hardware-Trojan classification method using machine learning at gate-level netlists based on Trojan features. IEICE Trans Fundam Electron Commun Comput Sci 100(7):1427–1438
12. Hasegawa K, Yanagisawa M, Togawa N (2017) Hardware Trojans classification for gate-level netlists using multi-layer neural networks. Proc. IEEE Symposium on On-Line Testing and Robust System Design (IOLTS), pp 227–232
13. Huang G, Gong Y, Xu Q, Wattanachote K, Zeng K, Luo X (2020) A convolutional attention residual network for stereo matching. IEEE Access 8:50828–50842
14. Inoue T, Hasegawa K, Kobayashi Y, Yanagisawa M, Togawa N (2018) Designing subspecies of hardware trojans and their detection using neural network approach. Proc. 2018 IEEE 8th Int Conf Consum Electron - Berlin (ICCE-BERLIN). Berlin, Germany
15. Jin Y, Makris Y (2008) Hardware Trojan detection using path delay fingerprint. Proc.2008 IEEE International Workshop on Hardware-Oriented Security and Trust. Anaheim, CA, pp 51–57
16. Jin Y, Sullivan D (2014) Real-time trust evaluation in integrated circuits. Proc.2014 Design, Automation & Test in Europe Conference & Exhibition (DATE). Dresden, Germany, pp 1–6
17. Jin Y, Maliuk D, Makris Y (2012) Post-deployment trust evaluation in wireless cryptographic ICs. Proc.2012 Design, Automation & Test in Europe Conference & Exhibition (DATE). Dresden, Germany pp 965–970
18. Kkalais (2020) Machine Learning Techniques for Hardware Trojan Detection, github.com. Available: https://github.com/Kkalais/Hardware-Trojan-Detection. Accessed 1 Oct 2022
19. Kulkarni A, Pino Y, Mohsenin T, "SVM-based real-time hardware Trojan detection for many-core platform," Proc. (2016) 17th International Symposium on Quality Electronic Design (ISQED). Santa Clara, CA, USA 2016:362–367
20. Liu M, Yu Y, Liao Q, Zhang J (2020) Histopathologic cancer detection by dense-attention network with incorporation of prior knowledge. Proc.2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI). Iowa City, IA, USA, pp 466–470
21. Liu P, Zhang C, Qi H, Wang G, Zheng H (2022) Multi-Attention DenseNet: a scattering medium imaging optimization framework for visual data pre-processing of autonomous driving systems. IEEE Trans Intell Transp Syst 23(12):25396–25407
22. Muralidhar N, Zubair A, Weidler N, Gerdes R, Ramakrishnan N (2021) Contrastive graph convolutional networks for hardware trojan detection in third party IP cores. Proc.2021 IEEE International Symposium on Hardware Oriented Security and Trust (HOST). Tysons Corner, VA, USA, pp 181–191
23. Nasr AA, Abdulmageed MZ (2016) Automatic feature selection of hardware layout: a step toward robust hardware trojan detection. J Electron Test 32(3):357–367
24. Nowroz AN, Hu K, Koushanfar F, Reda S (2014) Novel techniques for high-sensitivity hardware trojan detection using thermal and power maps. IEEE Trans Comput Aided Des Integr Circuits Syst 33(12):1792–1805
25. Rad RM, Wang X, Tehranipoor M, Plusquellic J (2008) Power supply signal calibration techniques for improving detection resolution to hardware Trojans. Proc.2008 IEEE/ACM Int Conf Comput Aided Des. San Jose, CA, USA, pp 632–639
26. Rozhin Yasaei (2022) Hardware Trojan Power & EM Side-Channel dataset, IEEE DataPort. Available: https://ieee-dataport.org/open-access/hardware-trojan-power-em-side-channel-dataset. Accessed 17 Oct 2022
27. Salmani H (2017) COTD: Reference-free hardware trojan detection and recovery based on controllability and observability in gate-level netlist. IEEE Trans Inf Forensics Secur 12(2):338–350
28. Salmani H, Tehranipoor M, Karri R (2013) On Design vulnerability analysis and trust benchmark development. Proc. 2013 IEEE 31st Int Conf Comput Des (ICCD), pp 471–474
29. Salmani H, Tehranipoor M, Plusquellic J (2010) A layout-aware approach for improving localized switching to detect hardware Trojans in integrated circuits. Proc.2010 IEEE International Workshop on Information Forensics and Security. Seattle, WA, USA, pp 1–6
30. Sankaran S, Mohan VS, Purushothaman A (2021) Deep learning based approach for hardware trojan detection. Proc. 2021 IEEE International Symposium on Smart Electronic Systems (iSES). Jaipur, India pp 177–182
31. Shakya B, He T, Salmani H, Forte D, Bhunia S, Tehranipoor M (2017) Benchmarking of hardware Trojans and maliciously affected circuits. J Hardw Syst Secur (HaSS)
32. Sharma R, Sharma GK, Pattanaik M (2021) A few shot learning based approach for hardware Trojan detection using deep siamese CNN. Proc.2021 34th International Conference on VLSI Design and 2021 20th International Conference on Embedded Systems (VLSID). Guwahati, India, pp 163–168

33. Stellari F, Song P, Weger AJ, Culp J, Herbert A, Pfeiffer D (2017) Verification of untrusted chips using trusted layout and emission measurements. Proc.2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST). Arlington, VA, USA, pp 19–24
34. Tang W, Su J, He J, Gao Y (2022) A deep learning method based on the attention mechanism for hardware trojan detection. Electronics, vol 11, no 15, pp 2400
35. Tehranipoor M, Karri R, Koushanfar F, Potkonjak M (2016) Trusthub. Available online: https://www.trust-hub.org
36. Tong W, Chen W, Han W, Li X, Wang L (2020) Channel-attention-based densenet network for remote sensing image scene classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13:4121–4132
37. Wang X, Salmani H, Tehranipoor M, Plusquellic J (2008) Hardware Trojan Detection and Isolation Using Current Integration and Localized Current Analysis. Proc.2008 IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems. Cambridge, MA, USA, pp 87–95
38. Woo S, Park J, Lee. JY, Kweon IS (2018) CBAM: Convolutional block attention module. Computer Vision – ECCV 2018. ECCV 2018, vol 11211, pp 3–19
39. Xie L, Huang C (2019) A residual network of water scene recognition based on optimized inception module and convolutional block attention module. Proc.2019 6th Int Conf Syst Informatics (ICSAI). Shanghai, China, pp 1174–1178
40. Xu Y, Chen Z, Huang B, Liu X, Dong C (2021) HTtext: A TextCNN-based pre-silicon detection for hardware Trojans. Proc.2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom). New York City, NY, USA, pp 55–62
41. Yu S, Gu C, Liu W, O'Neill M (2022) Deep learning-based hardware trojan detection with block-based netlist information extraction. IEEE Trans Emerg Top Comput 10(4):1837–1853

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Shouhong Chen received his Ph.D. degree in Mechanical and Electronic Engineering from Jiangsu University, Zhenjiang, Jiangsu, China, in 2021. He is currently a Professor at the School of Electronic Engineering and Automation, Guilin University of Electronic Technology, China. His current research interests include computer-aided testing, defect recognition, hardware Trojan detection, and deep learning.

Tao Wang was born in Yangzhou, China, in 1999. He is studying for a master's degree in electronic information at Guilin University of Electronic Technology. His current research interests include hardware security analysis and hardware Trojan.

Zhentao Huang was born in Hunan, China, in 1998. He is currently pursuing an M.S. degree in the School of Electronic Engineering and Automation at Guilin University of Electronic Technology, Guilin, China. His research interests include computer-aided testing, defect recognition, and deep learning.

Xingna Hou is a Ph.D. student from Guilin University of Electronic Technology. She is currently an associate professor with the School of Electronic Engineering and Automation, Guilin University of Electronic Technology, China. Her research interests include computer-aided testing, defect recognition, hardware Trojan detection, and deep learning.