sensors

MDPI

Article

Hausdorff Distance Model-Based Identity Authentication for IP Circuits in Service-Centric Internet-of-Things Environment†

Wei Liang [1,2], Weihong Huang [3], Wuhui Chen [4], Kuan-Ching Li [5,*] and Keqin Li [6]

1 School of Opto-Electronic and Communication Engineering, Xiamen University of Technology, Xiamen 361024, China; wliang@xmut.edu.cn
2 Key Laboratory of Fujian University of Internet of Things Applied Technology, Xiamen University of Technology, Xiamen 361024, China
3 College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China; whhuang@hnu.edu.cn
4 School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510275, China; chenwuh@mail.sysu.edu.cn
5 Department of Computer Science and Information Engineering, Providence University, Taichung 43301, Taiwan
6 Department of Computer Science, State University of New York New Paltz, New York 12561, USA; lik@newpaltz.edu
* Correspondence: kuancli@pu.edu.tw; Tel.: +886-4-2632-8001
† This paper is an extended version of our paper published in Huang, Y.; Liang, W.; Long, J.; Xu, J.; Li, K.-C. A Novel Identity Authentication for FPGA Based IP Designs. In Proceedings of the 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, 1–3 August 2018.

Received: 21 December 2018; Accepted: 21 January 2019; Published: 24 January 2019

Abstract: Rapid advances in the Internet-of-Things (IoT) have exposed the underlying hardware devices to security threats. As the major component of hardware devices, the integrated circuit (IC) chip also suffers the threat of illegal, malicious attacks. To protect against attacks and vulnerabilities of a chip, a credible authentication is of fundamental importance. In this paper, we propose a Hausdorff distance-based method to authenticate the identity of IC chips in IoT environments, where the structure is analyzed, and the lookup table (LUT) resources are treated as a set of reconfigurable nodes in field programmable gate array (FPGA)-based IC design. Unused LUT resources are selected for insertion of the copyright information by using the depth-first search algorithm, and the random positions are reordered with the Hausdorff distance matching function next, so these positions are mapped to satisfy the specific constraints of the optimal watermark positions. If the authentication process is activated, virtual positions are mapped to the initial key file, yet the identity of the IC designed can be authenticated using the mapping relationship of the Hausdorff distance function. Experimental results show that the proposed method achieves good randomness and secrecy in watermark embedding, as well the extra resource overhead caused by watermarks are promising.

Keywords: Hausdorff distance model; IP circuit; Internet of Things

1. Introduction

In recent years, with the widespread use and development of the Internet of Things (IoT), security issues on hardware have gained attention and wide concern [1,2]. With intellectual property

(IP) as the primary module in designing a complex system-on-chip (SoC), the reuse of IP can significantly reduce design and prototyping costs, shortening also the design cycle. Notwithstanding this, IP infringements frequently occur with the massive increase in the usage and quantity of devices and, thus, security primitives are implemented with increasing frequency for protection. For example, Castillo et al. [3] utilized the lookup tables (LUTs) for watermark embedding, where the copyright data is inserted into the space between the used and unused LUTs, and attackers are tough enough to remove since it is concealed into the functional resources of the design. Therefore, the authentication requires an extraction circuit to extract the secret data, causing substantial hardware overhead, and the redundant extraction circuit can also easily be attacked. Chen et al. [4] proposed a public IP authentication scheme, where a pseudo-random identity data is encrypted and inserted into the original design as specific constraints, due to its public capability to satisfy the security requirements and can be detected by graph coloring and Boolean satisfiability proof. Further research used the unused port of the used LUT for watermark insertion [5–9], so the watermarked LUTs are transformed into Random Access Memory (RAM) or Linear Feedback Shift Register (LFSR) for enhancing the ability against the removal attacks. If any infringement occurs, the copyright information can be detected from the bitfile of the design.

Nevertheless, the above methods easily leak the real positions of the watermarks, which exposes the copyright authentication to threat. To address this issue, Saha et al. [10] proposed to authenticate the IP identity by using a zero-knowledge proof protocol. Not leaking the really sensitive information, such as the watermark position or the watermarked content [11], can enable it to resist tampering attacks, but unfortunately the ability against removal attacks is also lower. If the watermark is impaired after being attacked, the copyright authentication may fail. Zhang et al. [12] used an obfuscation technique to enhance hardware security.

Some IP protection techniques are based on the encrypted bitfile, which requires extra decryption logic, and may cause an increase of the hardware resource and power consumption; for instance, the bitfile of a design implemented on a Static Ramdom Access Memory (SRAM) based field programmable gate array (FPGA) is stored in the external memory, e.g., Electrically Erasable Programmable Read—Only Memory (EEPROM). As the FPGA powers up, the bitstream is loaded into the FPGA device [13]. Depicted in Figure 1, the IP owner designed an IP core that propagates in an untrusted environment. In this case, the IP core may be attacked by illegal attackers. If the infringement occurs, the IP owner can authenticate the identity of the IP.

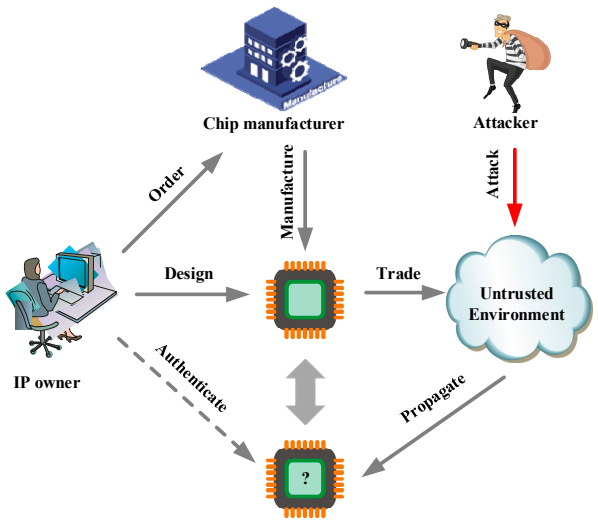


Figure 1. The structure of IP authentication system.

In recent years, many IP protection methods have been proposed to deter the infringement behavior [14–20] and have shown good performance to address security issues in IP protection.

Nevertheless, there are risks. The misappropriation of the kernel IP module causes significant economic loss to IP owners and developers.

In this work, a robust and secure authentication method is designed and proposed to authenticate the copyright of IP circuit chips in IoT environments. The Hausdorff distance shows excellent performance in fault tolerance and the ability of anti-interference, which is suitable for IP authentication. The theory of Hausdorff distance is introduced first and then an IP protection method proposed for the designed Hausdorff distance model, by which we could realize authentication using the non-similarity of two point-sets, M and N. If the isolated points in M are far away from N and also the matching degree of two sets is small, the larger values of unidirectional and bidirectional Hausdorff distances increase the similarity degree of two point-sets. The similarity between two IP designs is addressed by using the average Hausdorff distance, as the average Hausdorff distance considers the contribution from every point during the calculation of the Hausdorff distance.

The remainder of this paper is organized as follows. The constraint model based on Hausdorff distance is designed and described in Section 2, and the precise technique introduced for IP copyright authentication is presented in Section 3, where we concentrate on designing virtual node positions for inserting identification information. Experimental results are evaluated and compared to those from other techniques in Section 4, and finally, the paper is summarized as well as some future directions are included in Section 5.

2. Preliminaries

Digital IP circuit copyright authentication is a technique to securely protect the ownership of IP design. The IP designer inserts the copyright information in the circuit, which can indicate the identity of the designed IP cores. The inserted content of the copyright information is the binary information after encoding. When IP disputes occur, the designer can authenticate the original identity of the IP by extracting the inserted information in the circuit. In this work, a Hausdorff distance-based constraint function is proposed. The watermarks can be inserted into specific positions after virtual mapping. Within an accepted error range ε, the reverse method of mapping can be used to find the LUT resources which are relevant to the copyright information.

2.1. Intellectual Property (IP) Circuit Based on Hausdorff Distance

There are different types of IP cores at various design levels, as identification information can be inserted at any design level as a part of the IP core for protection [21,22]. FPGAs are programmable integrated circuits, and the process to insert identification information in such a circuit as a target is more complicated than that of traditional multimedia—text, software, among others. There are lots of configurable logic blocks (CLB) in an FPGA device, as each CLB includes several slices and some inner connections. LUTs in each slice can be configured as specific logic functions, and as a LUT is unused, we can insert specific data without affecting the performance of a design so that it can be used as the carrier of a watermark. Figure 2 shows the LUT configured as a logic function $f = \overline{AB} + AB\overline{CD} + ABCD$. With a specific input within 0~15 range, the corresponding result of the function is outputted.

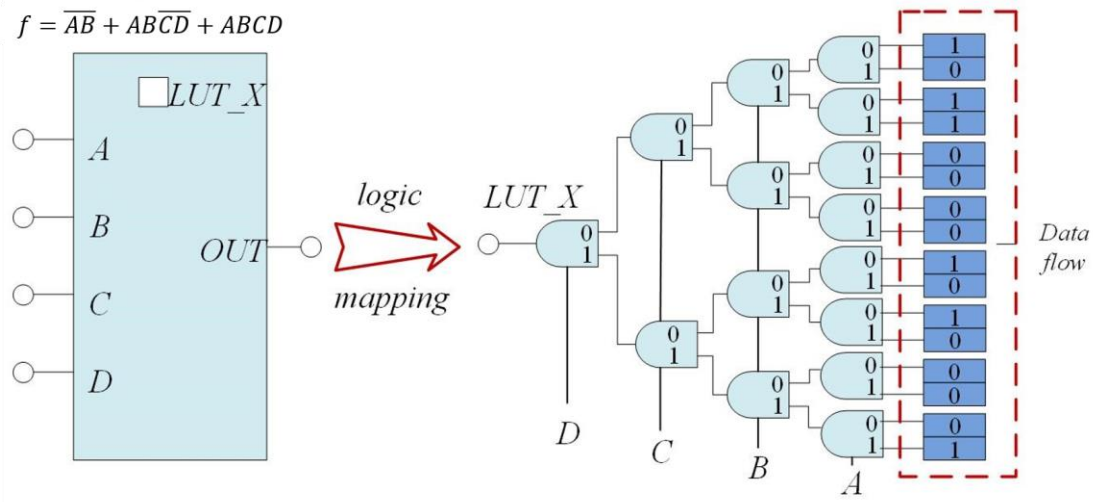


Figure 2. The structure of a lookup table (LUT) configured as a function $f = \overline{AB} + AB\overline{CD} + ABCD$.

Kahng et al. [23] proposed to hide the identification data into LUTs of FPGAs, as LUTs are ideal carriers due to the large number of LUTs in an FPGA-based IP core. A LUT can be treated as a RAM where data is written, as input generates an address to search the corresponding content which is the output. The recently used LUTs inter-relate to original LUTs with some "don't care" connections to enhance the security. In [24], the identification data is inserted into the STG (state transition graph) to authenticate the design copyright, which can be applied to the sequential design of a firm core. Such a technique inserts identification data in FSM (finite state machine) of IP design, which improves the robustness of the watermarks and can be safely implemented [25]. This was the first public IP watermarking technique developed at FSM level. Later, constraint-based watermarking techniques brought additional constraints into a watermarked design [26], which brought extra overheads and degraded the performance.

Notwithstanding this, the constraints added at higher level caused some unpredictable issues [27]. The primary purpose of IP protection methods is to prevent IP copyright in chips from being inappropriately accessed. In practice, the authentication system is implemented with passwords and key encryption, whose security is insufficient. Malicious attackers can perform deep attacks on the electronic chip, such as cloning or collusion attacks. If the chip has higher requirements on the security level, such a system cannot meet the demands [28].

To strengthen the security of IP authentication, Xu et al. [29] proposed a chaos mapping-based IP protection method with better reliability and security. However, this also leaks sensitive information of IP watermarks during authentication, such as position or content. Additionally, the constrained watermarks cannot be detected at another design level, where the low traceability is also a drawback. Cui et al. proposed an ultra-low overhead dynamic watermarking on scan design for hard IP protection [30], which has greatly improved the security of the watermarks and reduced the overhead.

The Hausdorff distance-based position selection algorithm is to search the unused LUTs around the original design and utilize them in inserting the identification data. Due to serious IP infringements in chip design, researchers exhibit high demands on IP identification techniques. As a popular IP protection technique, IP identification can provide copyright authentication and avoid IP infringement. In this work, random positions around the original design are utilized for identification yet provide real-time protection for IP cores, and the major advantage is the insertion of an enormous amount of identification data. Once attacked, the identification data can also be recovered with the created mapping relationship. Figure 3 shows the diagram of the Hausdorff distance-based IP authentication technique, whereas the IP owner inserts the processed ownership information into the bitfile core. Each IP design contains the information of different users, which is helpful support for tracing the infringement. Such an IP owner can authenticate the IP design with the Hausdorff Distance-based

technique. If the ownership information is successfully detected, the copyright is proven; otherwise, the IP design is embezzled.

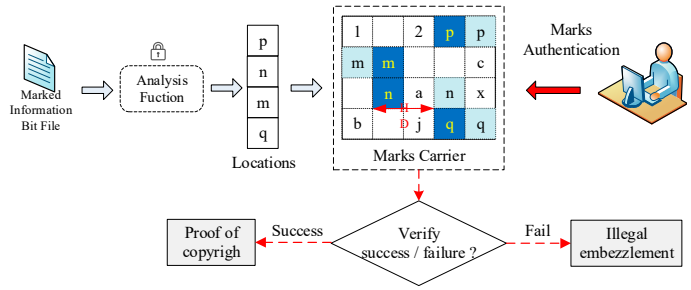


Figure 3. IP authentication based on the Hausdorff distance model.

2.2. Hausdorff Distance-Based Constraint Model

Hausdorff distance (HD) originated from the differential dynamic [31,32] that describes the similarity between two sets of points and widely used in fields of secure identification, since it requires calculating point-to-point distances. In this work, we define the maximum and minimum distances between two sets of IP position points. The maximum mismatching degree between two sets of the point is calculated, so a smaller HD value signifies that the sets are similar; otherwise, the sets have lower similarity. The definitions are as follows.

Definition 1. *Given two sets of point A={a_1, a_2, a_3, . . . , a_p} and B={b_1, b_2, b_3, . . . , b_q}, the Hausdorff Distance-based constraint between A and B is defined as follows.*

$$H(A,B) = \max(h(A,B), h(B,A)) \tag{1}$$

$$h(A,B) = \max_{a\in A} \min_{b\in B} \|a-b\| \tag{2}$$

$$h(B,A) = \max_{b\in B} \min_{a\in A} \|b-a\| \tag{3}$$

As in (2) and (3), *h(A,B)* is the directed Hausdorff distance from *A* to *B*, and *h(B,A)* is the directed Hausdorff distance from *B* to *A*, with $\| \ \|$ as the norm distance of *A* and *B*, *h(A,B)* is seen as the maximum value of distances of each point in *A* to *B*, and *h(B,A)* the maximum distance value of points in *B* to *A*. From this, HD is obtained from the maximum mismatching degree between *A* and *B* by calculating the maximum value of *h(A,B)* and *h(B,A)*.

Definition 2. *If A and B satisfy affine transformation relationship, $HF_a(x_a,y_b)$ and $HF_b(x_b,y_b)$ are two points respectively in A and B, the mapping relationship is defined as (4)*:

$$\begin{cases} HF_a = m_{00}x_a + m_{01}y_a + l_x \\ HF_b = m_{10}x_a + m_{11}y_a + l_y \end{cases} \tag{4}$$

Here, l_x and l_y are, respectively, the offset in x and y directions, and the vector HF_{ab}=(m_{00},m_{01},m_{10},m_{11},l_x,l_y) denotes the matching relationship. Based on Definition 1, the transformation of Hausdorff distance at two sets of points in LUTs is defined as (5), deriving (6) and (7).

$$H_{LK}(A,B) = \max(\mathrm{h}_L(A,B), h_K(B,A)) \tag{5}$$

$$\mathrm{h}_L(A,B) = L^{th}_{a_i\in A} \min_{b_j\in B} \|a_i - b_j\| \tag{6}$$

$$h_K(B,A) = L^{th}_{b_j \in B} \min_{a_i \in A} \|b_j - a_i\| \tag{7}$$

In Equation (6), $L^{th}_{a_i \in A}$ represents the L-th value of the unidirectional Hausdorff distances from A to B in descending order, $K^{th}_{b_j \in B}$ the K-th value of the unidirectional Hausdorff distances from B to A in descending order. As $K = L = 1$, Equation (5) is the original Hausdorff distance from A to B. To eliminate the interference of isolated points, the sensitivity of Hausdorff distance is reduced. However, there exists the case that HD cannot accurately describe the similarity between point sets. Thus, average Hausdorff distance is applied in (9) to address issues of similar type.

$$H_{mean}(A,B) = \max(\mathrm{h}_{mean}(A,B), h_{mean}(B,A)) \tag{8}$$

Here, we have:

$$\mathrm{h}_{mean}(A,B) = \frac{1}{N_A} \sum_{a_i \in A} \left(\min_{b_j \in B} \|a_i - b_j\| \right) \tag{9}$$

$$h_{mean}(B,A) = \frac{1}{N_B} \sum_{b_j \in B} \left(\min_{a_i \in A} \|b_j - a_i\| \right) \tag{10}$$

In Equation (9), N_A is the number of characteristic points in A, and N_B in (10) the number of characteristic points in B. The average Hausdorff distance considers the contribution of each point in calculating the Hausdorff distance and, therefore, it is real and of higher accuracy than the description of the similarity degree between two points of the set.

$$H\left(t_{affine}[A], B\right) = \max\left(\mathrm{h}\left(t_{affine}[A], B\right), h\left(B, t_{affine}[A]\right)\right) \tag{11}$$

The Hausdorff distance coefficient represents the matching degree between the positions of identification data and the constraint function. In the case with lower requirements for accuracy, it can reduce the computation complexity without affecting the performance by setting the small component of Hausdorff distance coefficient to zero. Based on this fact, the component of the Hausdorff distance coefficient with the most considerable absolute value will be regarded as the position constraint characteristic of the IP circuit, and the high-frequency coefficient in the position constraint function of IP circuit is set to zero. After that, it will be reconfigured. The maximum absolute value of the Hausdorff distance coefficient is calculated as the candidate position characteristic vector. With principal component analysis (PCA) and normalization processing, the position characteristic vector between two points is obtained, so the collection of position characteristic vectors is successfully extracted.

Assuming that the response signal of the IP circuit is resolved by the position constraint function, the a-th component of the b-th Hausdorff distance coefficient is denoted by h^b_a $(a = 1, 2, 3, \ldots, N)$. Consequently, the candidate position characteristic H_b is defined as (12).

$$H_b = \max\left(\left|h^b_a\right|\right), a = 1, 2, \cdots, M \tag{12}$$

where M is the dimension of the Hausdorff distance coefficient. The candidate position characteristic vector is denoted as (13) and (14).

$$HW_x = \varepsilon[x_1, x_2, \cdots, x_a, \cdots x_N]^T \tag{13}$$

$$HW_y = \varepsilon[y_1, y_2, \cdots, y_b, \cdots y_N]^T \tag{14}$$

The position searching function HS can be used to search exact positions of related points, and is represented as follows:

$$HS_x = \min H_{ab}(U, V, W) = \sum_{a=1}^{N} \sum_{b=1}^{N} u_{ab}^m \|x_b - x_a\|^2 \tag{15}$$

The constraint is described as (16):

$$\begin{array}{c} \forall a \in \{1, 2, \cdots, N\},\ b \in \{1, 2, \cdots, N\} \\ u_{ab} \in [0, 1],\ \sum_{a=1}^{N} u_{ab} = 1,\ 0 < \sum_{b=1}^{N} u_{ab} \leq N \end{array} \tag{16}$$

In (15), U is the position collection of LUT array, V is the position collection array of the center, W is the collection of identification data, c is the number of clusters, $x_b \in R_p$ denotes the b-th data pattern, and u_{ab} is the position degree of x_b belonging to the b-th category.

The position constraint function in (15) is normalized, yielding the position characteristic vector of Hausdorff distance. The normalized position constraint function is described as (17):

$$\begin{aligned} HS_{AB} &= \min_{(U,V,W)} H_{ab}(U, V, W) \\ &= \sum_{a=1}^{N} \sum_{b=1}^{N} u_{ab}^m \|x_b - v_b\|^2 + \frac{\beta}{m^2 \sqrt{c}} \sum_{a=1}^{N} \sum_{b=1}^{N} \left(u_{ab}^m \log u_{ab}^m - u_{ab}^m\right) \end{aligned} \tag{17}$$

3. Hausdorff Distance-Based Authentication Algorithm

This section introduces the Hausdorff distance-based authentication algorithm, and the symbols used are as follows: $H(A, B)$ is the bidirectional Hausdorff distance, and $h(A, B)$ and $h(B, A)$ are the unidirectional Hausdorff distance from A to B and B to A respectively. The distance of two points, a in A and b in B, can be calculated by $a - b$. The calculated distances are ordered to get the maximum one, denoted by $h(A, B)$, and similarly the $h(B, A)$. The value of $H(A, B)$ is the highest value of $h(A, B)$ and $h(B, A)$, denoting the non-similarity degree of two point-sets. Thus, the position constraint function is analyzed with Equation (11). The main steps of IP authentication are illustrated, including position selection, identification data insertion, identification matching, and copyright authentication.

3.1. Position Selection

In this work, a real-time IP authentication algorithm based on the Hausdorff distance constraint model is proposed to insert the identification data into IP circuits effectively. This searches the unused LUTs from the FPGA-based IP design, as well those secure positions analyzed to find out those which cannot be attacked. From (21), it is defined as a position constraint factor ε to constrain the positions of identification data, whose condition is that the similarity of Hausdorff distance represents the difference of IP circuits.

The design of a position selection algorithm to search suitable positions for inserting identification data is presented next, and the steps are as follows. First, a search to unused LUTs with the length of identification data is processed, and the similarity between the original IP design and the watermarked IP design is calculated by the Hausdorff distance model next. The value of Hausdorff distance is zero if both circuits match to be the same. Though, due to the insertion of the identification data, the value will not be zero. A matching error η exists in the calculation of Hausdorff distance, so it should be reduced to an acceptable range ε. The identification data is recorded in a key file for better security. Besides, the position mapping algorithm is utilized to prevent the key file from being leaked. The Hausdorff distance-based authentication method can determine the mapping relationship between the positions of the original design and the watermarked design. If the matching error falls in the range ε, then the selected positions for identification data insertion are secure; otherwise, the positions

should be selected again by using the above steps. The pseudo-code of the algorithm is described in Algorithm 1.

Algorithm 1: IP Marks embedding the algorithm

Input: (i) Position P_n (ii) Marks sequence W
Output: (i) Pseudo position L_n (ii) Constraint function HW_{xy}
/*Marks sequence $W = \{w_0, w_1, w_2, \ldots, w_n\}$, L means that P after the Marks sequence is embedded, the virtual position generated by the constraint HW_{xy}*/
/* C_n represent unused resource sequence C_n*/
1: Single location information P_i selected from C_n;
2: Calculate HW_{xy} by Constraint function HF_{ab} and P_i;
3: Calculate Single pseudo location information L_i by Constraint function HW_{xy} and P_i;
4: If $L_i \in C_n$ then
5. Store location information P_i and L_i;
6: Else if $L_i \notin C_n$ then
7: Delete the element P_i from the collection P_n;
8: End If
9: Store P_i in P_n;
10: For i:= 0 to n
11: Embed w_i into Location P_i in P_n;
12: End For
13: Output L_n and HW_{xy};

3.2. Position Characteristic Matching

(1) Position selection. LUT resources in FPGA can be represented as a *M*N* array, and the position of a LUT is given by (x_a, y_b). The product of area errors can provide the position matching value, and the similarity is expressed by (18).

$$SAD(u,v) = \iint_{\Phi} |f_1(x,y) - f_2(x+u, y+v)| dxdy \tag{18}$$

Equation (18) is normalized to generate (19). Hereafter, f_1 and f_2 are inserted and extracted identification data, respectively.

$$SAD(u,v) = \iint_{\phi} |(f_1(x,y) - \overline{f_1}) - (f_2(x+u, y+v) - \overline{f_2}(u,v))| dxdy \tag{19}$$

The position characteristic value of the Hausdorff distance is generated after PCA analysis and normalization. With (19), the characteristic matching value SAD is obtained.

HD matching method is to match the IP circuit including the selected positions of identification data with the one that includes the unselected positions of identification data. The most suitable initial positions of identification data are found within an acceptable range ε, denoted by LOC1.

(2) Position mapping. The positions selected by (1) should be further mapped with Definition 2, and the distribution of the identification data is shown in Figure 4. Selected LUTs with Equation (8) are different from the initially selected positions. The Hausdorff distance matching algorithm is used to verify the mapped positions of the identification data. If the Hausdorff distance falls into the range ε, the set of the mapped positions is denoted by LOC2. Otherwise, the selected positions cannot be used for identification data insertion and should be selected again.

(3) Identification data insertion. Detailed steps of insertion are as follows.

1. Generate the IP design for identification data insertion. The integrated circuit is implemented using design tools such as ISE or Quartus, Modelsim and Synplify synthesis tool for further implementation. The bitfile design is generated for a specific FPGA device.
2. Generate the identification data. The insertion procedure is illustrated by an example of inserting two identification data, M1 and M2, that should be transformed into binary data at first.
3. Search the positions for inserting identification data by traversing the point set and indexes ascendingly. The binary data is orderly inserted into the selected positions. With the traversing algorithm, the characteristic value of the selected position can be calculated. Meanwhile, it can be regarded as the clue to determine the virtual positions, whereas the positions are stored with tree indexes. Besides, the priority queue is used to control the accessed sub-nodes.

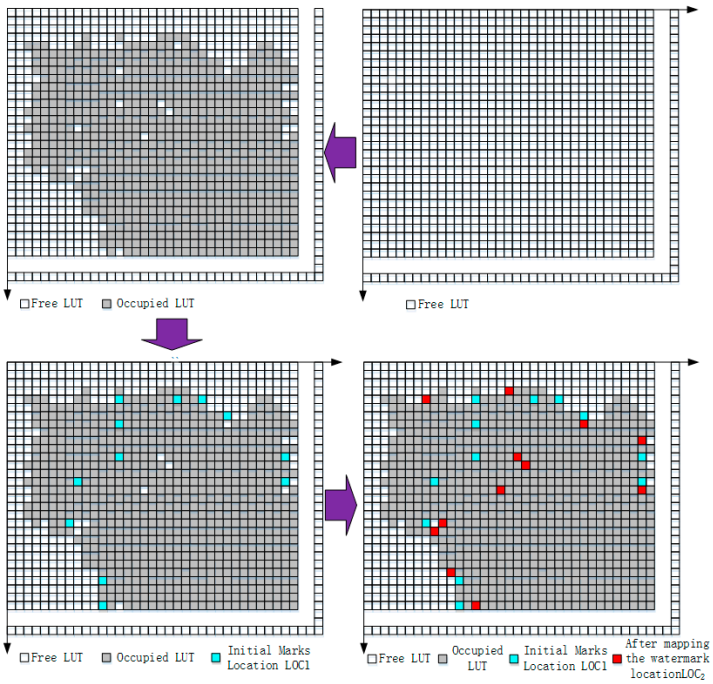


Figure 4. The distribution of identification data.

A. Classifying the positions of LUTs with scanning algorithm, i.e., used and unused LUT collection. In the unused collection, the LUTs with the number equal to the fragments of the identification data are selected. The Hausdorff distance is applied to calculate the characteristic matching degree between the IP cores that includes the selected positions and those with the unselected positions. Within the constraint range ε, a position collection for inserting identification data can be determined and denoted by LOC1. LOC1 is stored in a key file, and the position mapping algorithm is further mapped to get virtual position collection LOC2. In this case, the real positions are covered to ensure better security. Finally, Hausdorff distance is used to verify LOC2, if the verification result is within the range of ε. If positive, the mapping will be successful; the positions should be selected again if otherwise. The position mapping can avoid the attackers obtaining the real positions of identification data from the key file. Thus, the stored position collection LOC1 in the key file is not the real position to insert the identification data.

B. Modify or replace the control state of LUT resources. The generated identification data can be inserted into the selected secure positions, as some extra connections are added to make the newly added resources integrate with the functional resources, and potentially enhance the security of the identification data. The identification data matching algorithm is depicted in Algorithm 2.

Algorithm 2: IP Marks matching algorithm

Input: (i) Pseudo Position L_n (ii) Marks sequence W'
Output: (i) Current design L'_n
/*Marks sequence $W\prime = \{w\prime_0, w\prime_1, w\prime_2, \ldots, w\prime_n\}$, Current design L'_n means that the location of the information for embedding the Marks */
/*Compare the size of L_n and W'*/
1: gets(L_n); gets(W');
/*Traversing each element of the collection L_n and the collection W'*/
2: for (i=0; $L[i]$!='\0'&&$W'[i]$!='\0'; i++);
3: if ($L[i] \neq W\prime[i]$)
4: break;
5: else if $L[i] = W'[i]$
6: Embed $W'[i]$ into Location $L[i]$ in L_n;
7: Store $L[i]$ in L'_n;
8: End If
9: End For
10: Output L'_n;

3.3. Authentication of Identification Data

If IP infringement occurs, the authentication process is activated, and the authentication algorithm is processed in four steps: identification data locating, matching, data combination, and authentication, as shown in Figure 5.

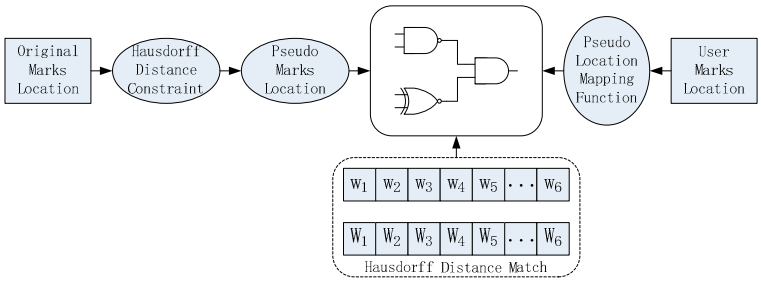


Figure 5. The detailed authentication flow chart.

(1) Identification data extraction. The virtual position set LOC2 can be determined with LOC1 recorded in the key file and the mapping algorithm. For the bitfile design or the physical layout, the fragments of the identification data can be extracted from specific LUTs.

(2) Fragments combination. In step 1, the fragments are extracted and ordered to acquire two binary sequences that include the identification data and the address relation factor. Therefore, the sequences should be split to retrieve the original watermarked positions. Based on the reverse procedure of insertion, the sequence should be divided by the same length to obtain the identification data fragments. These fragments are then combined to get the original positions and the virtual positions.

(3) Decryption and authentication. The stored key can decrypt virtual positions, and after that, the generated binary sequence is also transformed into plain text, as well the decrypted and declared data perform similarity matching operations. If the similarity degree is high enough, it will prove the legality of IP copyright.

The authentication algorithm of identification data is described in pseudo-code depicted in Algorithm 3.

Algorithm 3: IP Marks Authentication algorithm

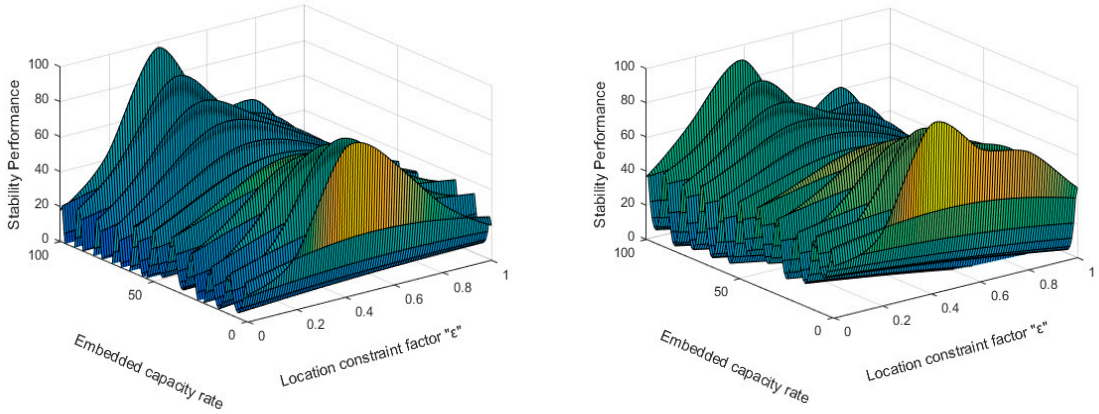
Input: (i) Position P_n and L'_n (ii) Constraint function HW_{xy}
Output: (i) Legal user (ii) Illegal user
/* To satisfy the condition HW_{xy}, L'_n and P_n are compared, and then judge whether L'_n is legal or not */
/* L''_n represents the element that satisfies the condition of HW_{xy},S_n represents the same element */
1: Calculate L''_n according to L'_n and HW_{xy}
/* Compare each of S_n and P_n elements */
2: gets(L''_n); gets(P_n);
for (i = 0; i < sizeof(L''_n) / sizeof($L''_n[0]$); i++)
{ for (j = 0; j < sizeof(P_n) / sizeof($P_n[0]$); j++)
{
if (L''_n[i]==P_n[i])
{
Store L''_n in S_n;
}
}
}
3: End for
/* Evaluate the similarity */
4: If (sizeof(S_n) / sizeof(P_n) $\geq$ 0.8)
5: Output: Legal user
6: else if (sizeof(S_n) / sizeof(P_n) < 0.8)
7: Output: Illegal user;

4. Experimental Results and Analysis

We have conducted experiments to evaluate the performance of the proposed method. The IP cores, such as DES, Cache, and RS, are implemented on XC2V800 FPGA device (Xilinx Inc., San Jose, CA, USA), and the tools utilized for evaluation include ISE, Modelsim, and Synplify. The performance evaluations focused on the stability, resource overhead, anti-attack ability, and similarity.

4.1. Stability Evaluation

Regarding stability, Hausdorff distance is used to determine the virtual positions for identification of data insertion. Under the control of coefficient ε, virtual positions will be distributed among the functional resources. This addresses the issue that attackers can find the real positions of identification data by analyzing attacks. Besides, the initial positions are further mapped to ensure security. The initial positions are stored in the key file, despite the real positions for inserting the identification data are the mapped ones. Even if the attacks retrieve the key file and extract the initial positions, it cannot determine the real positions of identification data. The Hausdorff distance constraint function is used to match the LUTs and make the inserted identification date secure. As the copyright needs authentication, the user can apply for the authentication parameters from the IP owner and perform authentication using the parameters. Therefore, even if the identification data is successfully analyzed and tampered by the attackers, it cannot pass the matching verification. As shown in Figure 6, we set two attack strengths, G = 20 and G = 40, where the stability performance is gentle, demonstrating the validity of the proposed scheme.



(a) Stability performance: G = 20 (b) Stability performance: G = 40

Figure 6. The stability performance evaluation.

4.2. Resource Overhead

Extra resource overhead after identification data insertion is evaluated and compared to three methods, as listed in Table 1. Four IP circuits are selected in this experiment, respectively Audio, DES, RS, and Cache. The proposed scheme utilizes the unused resources for identification data insertion, and the occupied resources slightly increase. Herein, "Time" represents the overall time to complete the authentication.

The proposed method can authenticate the copyright in real-time. By contrast with previous works where schemes authenticate the identity of the IP circuit by using multiple rounds of inquiry and the real-time authentication is affected, operations are involved in each round of authentication. In this work, the authentication is based on the Hausdorff distance model, and matching operation based on Hausdorff distance utilized for position calculation. The random positions are ordered and used to realize position mapping, as virtual positions satisfy the constraint condition of the optimal collection of positions. With the authentication activated, the virtual positions in the key file are used and decrypted for copyright authentication. In this section, we evaluate the overhead of the proposed scheme and the comparative schemes [28–30]. In [28], the authors proposed a robust low overhead IP watermarking algorithm. It compressed the real watermarks, which greatly reduced the overhead. The use of position mapping can enhance the security of the inserted watermarks. In [29], the authors proposed a high polymetric mutual mapping IP watermarking algorithm. This uses the principle of the secret segmentation mechanism to build a mutual mapping connection between two watermarks. The algorithm can resist the removal attacks. The authors in [30] utilized the structure of scan design and realized an ultra-low overhead watermarking scheme. The result of overhead evaluation is shown in Table 1, and the operation time for the comparative schemes is larger than the proposed schemes. Also, the authentication efficiency is better than similar schemes.

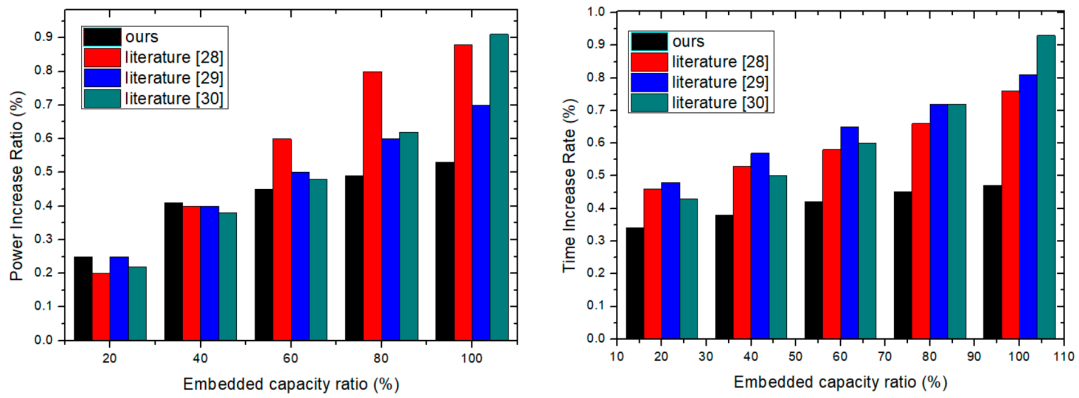
Table 1. Overhead evaluation for four types of intellectual property (IP) circuits.

IP Circuit	Occupied Resources	Algorithm	Average Hausdorff Distance	Time (ns)	ε
Audio	424	Literature [30]	-	12.77	0.423
		Literature [29]	-	10.56	0.439
		Literature [28]	-	16.56	0.487
		ours	14.43	7.64	0.439
DES	7064	Literature [30]	-	12.48	0.437
		Literature [29]	-	12.16	0.498
		Literature [28]	-	12.15	0.436
		ours	17.53	5.13	0.427

Table 1. *Cont.*

IP Circuit	Occupied Resources	Algorithm	Average Hausdorff Distance	Time (ns)	ε
RS	7392	Literature [30]	-	13.57	0.473
		Literature [29]	-	12.49	0.416
		Literature [28]	-	13.67	0.435
		ours	13.42	6.84	0.484
Cache	14352	Literature [30]	-	12.29	0.469
		Literature [29]	-	14.3	0.468
		Literature [28]	-	17.82	0.479
		ours	15.74	5.87	0.467

In the proposed scheme, original identification data is encrypted using a hash function that compresses data into a message with the length of 128 bits, and the message is inserted into the design after being divided into a few fragments next. In this experiment, the power and time are evaluated with the increase of inserting identification data from 4 bits to 128 bits. The resource overhead is shown in Figure 7, where Figure 7a shows the time increase rate and Figure 7b the power increase ratio. With the increase of the embedded capacity ratio, the overhead curves of the three schemes are ascending, so the increase in time and power for the proposed scheme is the minimum (indicated as "ours").



(**a**) Time evaluation (**b**) Power evaluation

Figure 7. Time and power overhead evaluation with the increase of embedded capacity.

Figure 8 shows the rate of resource change. From this figure, the increase in resources is the lowest by comparison with the other three methods.

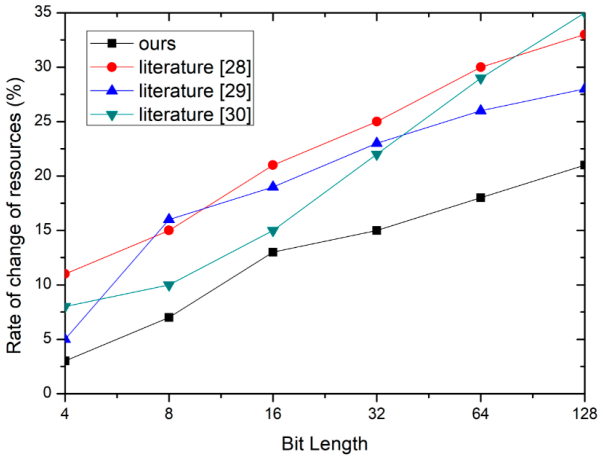
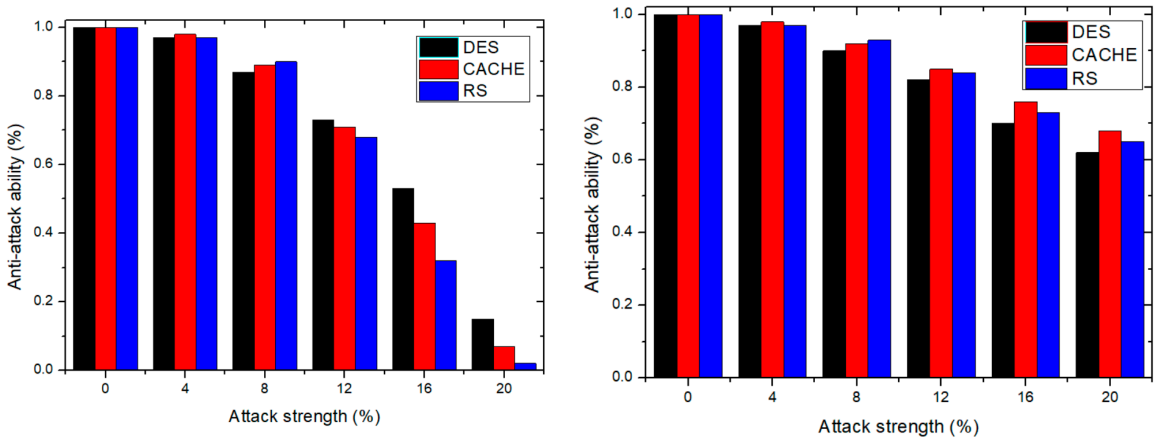


Figure 8. The evaluation of resource overhead.

4.3. Anti-Attack Ability

Several attacks threaten the security of IP design and IP watermarks. In this section, the replay attack is mainly considered to evaluate the security of position selection. First, the selected IP designs are implemented in the Xilinx ISE tool, with the identification data inserted into the designs of the proposed method. The generated design is the watermarked core, and attacked for performance evaluation purposes. In Figure 9, the X-axis denotes the attack strength, while the Y-axis denotes anti-attack ability. Figure 9a is the anti-attack ability of [30] and Figure 9b the proposed method. Results show that the positions of the identification data achieve better performance than [30]. When the attack strength is 20%, the anti-attack ability of our method exceeds 60% for three different cores. However, the method [30] shows a sharp decline when the attack strength exceeds 10%, showing the anti-attack ability of the proposed method to be encouraging and promising.



(**a**) Anti-attack ability of scheme in [30] (**b**) Anti-attack ability of the proposed scheme

Figure 9. The evaluation and comparison of the anti-attack ability.

The anti-attack ability of the proposed scheme to the other three comparative schemes is also compared, as shown in Table 2. Xu et al. proposed a distributed data hiding scheme that cannot resist physical attacks, replay attacks, and machine learning attacks [29]. Cui et al. proposed a scheme that can resist replaying attacks although it can do nothing against physical attacks, machine learning attacks and the false attacks [30]. The scheme proposed by Long et al. has the ability against replay attacks and fake attacks, although it cannot resist physical attacks and machine learning attacks [28]. Finally, the proposed scheme has the ability against all former types of attacks.

Table 2. The comparison of the anti-attack ability.

Program	Replay Attack	Physical Attack	Machine Learning Attack	Fake Attacks
Xu	No	No	No	Yes
Cui	Yes	No	No	No
Long	Yes	No	No	Yes
Ours	Yes	Yes	Yes	Yes

4.4. Similarity Evaluation

In order to reflect the effect of identification data on the functionality of the IP circuit, the location conversion coefficient HS_{ab} is utilized to evaluate the secrecy of identification data. This can be defined as the quadratic sum of the difference between the inserted and the extracted identification data, as defined in (20).

$$HS_{ab} = \min_{(u,v,w)} H_{ab}(u,v,w) = \sum_{a=1}^{N}\sum_{b=1}^{N} \|x_a - u_a\|^2 + \|y_b - v_b\|^2 \tag{20}$$

At this point, (u,v) represents the LUT positions of the IP circuit and, respectively, m and n are the numbers of rows and columns of the position array. A larger value HS_{ab} represents better security after inserting the identification data. Experiments show that it is difficult to analyze the difference between the original circuit and the marked circuit using the logic analyzer when HS_{ab} is greater than 30 dB. In this work, we utilize the method by combining Hausdorff distance and normalization parameter to evaluate the similarity of the inserted and extracted data. The similarity is better when the normalization parameter is more significant. The normalized parameter H_{nc} can be calculated by (21).

$$H_{nc} = \frac{\sum_{x=0}^{m-1}\sum_{y=0}^{n-1} f_1(x,y) \times f_2(x,y)}{\sqrt{\sum_{x=0}^{m-1}\sum_{y=0}^{n-1} f_1(x,y)}\sqrt{\sum_{x=0}^{m-1}\sum_{y=0}^{n-1} f_2(x,y)}} \tag{21}$$

In (21), f_1 and f_2 denote the inserted and the extracted identification data respectively, also M and N are respectively the bit lengths of both identification data. In the ideal case, the normalization parameter $H_{nc} = 1$. As shown in Figures 10 and 11, with the increase of the bit length, the HS_{ab} and H_{nc} are larger than that of the similar schemes, which demonstrates the excellent performance of the proposed scheme.

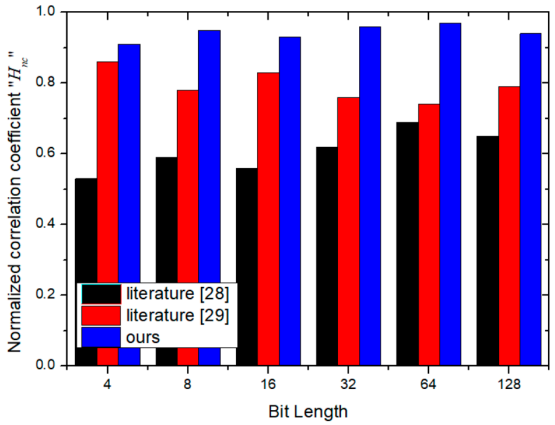


Figure 10. The evaluation of the normalized correlation coefficient.

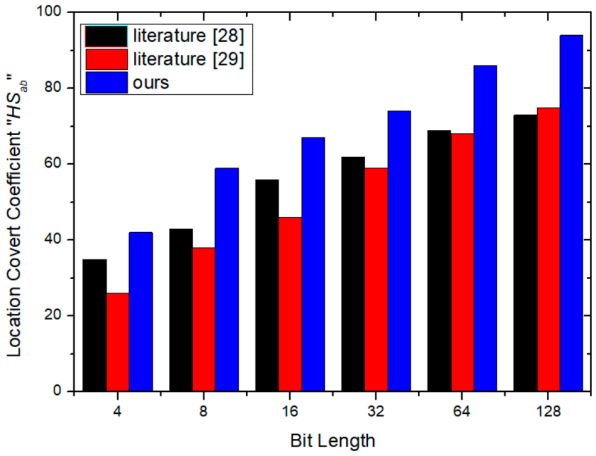


Figure 11. The evaluation of location covert coefficient.

5. Conclusions and Future Work

With the rapid advances of highly sophisticated electronics and chips in IoT devices and increasing industrial and social applications, IP authentication attracts more attention to malicious attacks. Chips must be verified and secure protected as thoroughly as designs for any other applications in IoT environments, as the high volumes expected for several different types of IoT devices mean that any vulnerability to such chips would be enormously expensive. In recent years, several methods have been proposed for IP protection, although some methods are complicated in copyright authentication after suffering attacks. In this paper, the main contributions are twofold: (1) a position mapping algorithm based on Hausdorff distance is proposed to generate the virtual positions, used to determine the initial positions for inserting the identification data and storing it in the key file. During authentication, the virtual positions can be found through the key file and Hausdorff distance function. By matching the characteristic, the real positions of identification data can be found for authentication, and (2) a method to compute the Hausdorff distance is proposed. According to the position characteristic, an optimization model is created, where the maximum Hausdorff distance is selected as the unidirectional distance among LUT resources. The proposed method could transform the computation of Hausdorff distance into the minimum distance between two points. As a result, the efficiency and stability are improved, shown and validated the proposed method, which has good performance for secured insertion of identification data.

Furthermore, the anti-attack ability is also enhanced. Even though some abecedarian researches on the Hausdorff distance-based IP authentication technique have been attempted, there are still some challenges. The proposed scheme mainly utilizes the distance as a measurement of similarity between two positions, and the position matching by the distance between LUTs should require the participation of the credible third party. Nevertheless, a full, credible third party may not exist in the industry community, although we assumed the existence of the credible third party. In this way, the IP authentication technique requires further investigation in practicability, security, and reliability.

Author Contributions: W.L. and W.H. conceived and designed the experiments; W.L., W.H. W.C. and K.-C. Li performed the experiments and analyzed the data; W.L., W.H. and W.C. wrote the paper under the supervision of K.-C.L. and K.L., who reviewed and edited.

Funding: This research was funded by the National Natural Science Foundation of China (Grant 61572188), Xiamen science and technology Foundation (Grant 3502Z20173035), Scientific Research Program of New Century Excellent Talents in Fujian Province University, Industrial Robot Application of Fujian University Engineering Research Center, Minjiang University (MJUKF-IRA201802), Fujian Provincial Natural Science Foundation of China (Grant 2018J01570) and the CERNET Innovation Project (Grant NGII20170411).

Acknowledgments: The authors would like to thank anonymous reviewers for their helpful advice on various technical issues examined and comments.

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

References

1. Lee, J.W.; Lim, D.; Gassend, B.; Suh, G.E.; van Dijk, M.; Devadas, S. A technique to build a secret key in integrated circuits for identification and authentication applications. In Proceedings of the 2004 Symposium on VLSI Circuits, Digest of Technical Papers, Honolulu, HI, USA, 17–19 June 2004; pp. 176–179.
2. Klapproth, P. General architectural concepts for IP core re-use. In Proceedings of the ASP-DAC/VLSI Design 2002 7th Asia and South Pacific Design Automation Conference and 15h International Conference on VLSI Design, Bangalore, India, 11 January 2002; p. 325.
3. Castillo, E.; Meyerbaese, U.; Parrilla, L.; Garcia, A.; Lloris, A. New advances for automated IP soft-core watermarking. *Proc. SPIE* **2009**. [CrossRef]
4. Chen, X.; Qu, G.; Cui, A. Practical IP watermarking and fingerprinting methods for ASIC designs. In Proceedings of the IEEE International Symposium on Circuits and Systems, Baltimore, MD, USA, 28–31 May 2017; pp. 1–4.

5. Liang, W.; Wu, K.; Zhou, H.; Xie, Y. TPCM: An IP Watermarking Algorithm based on Two Dimensional Chaotic Mapping. *Comput. Sci. Inf. Syst.* **2015**, *12*, 823–841. [CrossRef]
6. Abdel-Hamid, A.T.; Tahar, S.; Aboulhamid, E.M. A Survey on IP Watermarking Techniques. *Des. Autom. Embed. Syst.* **2004**, *9*, 211–227. [CrossRef]
7. Echavarria, J.; Morales-Reyes, A.; Cumplido, R.; Salido, M.A. FSM merging and reduction for IP cores watermarking using Genetic Algorithms. In Proceedings of the International Conference on Reconfigurable Computing and FPGAs, Cancun, Mexico, 8–10 December 2014; pp. 1–7.
8. Abbas, Y.A.; Jidin, R.; Jamil, N.; Z'aba, M.R.; Rusli, M.E. PRINCE IP-core on Field Programmable Gate Arrays (FPGA). *Res. J. Appl. Sci. Eng. Technol.* **2015**, *10*, 914–922. [CrossRef]
9. Xu, J.; Long, J.; Peng, L. A scattered IP watermarking algorithm in FPGA design. *J. Comput. Res. Dev.* **2013**, *50*, 2389–2396.
10. Saha, D.; Kolay, S.S. Secure public verification of IP marks in FPGA design through a zero-knowledge protocol. *IEEE Trans. VLSI Syst.* **2012**, *20*, 1749–1757. [CrossRef]
11. Lach, J.; Mangione-Smith, W.H.; Potkonjak, M. Robust FPGA intellectual property protection through multiple small watermarks. In Proceedings of the Design Automation Conference, Orleans, LA, USA, 21–25 June 1999; pp. 831–836.
12. Zhang, J.L.; Lin, Y.P.; Lyu, Y.Q.; Wang, X.Q. A chaotic-based publicly verifiable FPGA IP watermark detection scheme. *Sci. Sin. Inf.* **2013**, *43*, 1096–1110.
13. Ebrahimi, M.; Rao, P.M.B.; Seyyedi, R.; Tahoori, M.B. Low-Cost Multiple-Bit Upset Correction in SRAM-Based FPGA Configuration Frames. *IEEE Trans. VLSI Syst.* **2016**, *24*, 932–943. [CrossRef]
14. Cui, A.; Chang, C.H.; Tahar, S. A robust FSM watermarking scheme for IP protection of sequential circuit design. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2011**, *30*, 678–690. [CrossRef]
15. Liang, W.; Sun, X.; Xia, Z.; Sun, D.; Long, J. A Chaotic IP Watermarking in Physical Layout Level Based on FPGA. *Radioengineering* **2011**, *20*, 118–125.
16. Liang, W.; Zhang, D.; You, Z.; Li, W.; Bi, X.; Hu, Y. A Digital IP Watermarking Scheme of Based on Self-Recovery Secret Information. *J. Comput. Theor. Nanosci.* **2014**, *11*, 1727–1731. [CrossRef]
17. Pappu, R.; Recht, B.; Taylor, J.; Gershenfeld, N. Physical one-way functions. *Science* **2002**, *297*, 2026–2030. [CrossRef] [PubMed]
18. Lim, D.; Lee, J.W.; Gassend, B.; Suh, G.E.; van Dijk, M.; Devadas, S. Extracting secret keys from integrated circuits. *IEEE Trans. VLSI Syst.* **2005**, *13*, 1200–1205.
19. Liang, W.; Sun, X.; Ruan, Z.; Long, J.; Wu, C. A Sequential Circuit-Based IP Watermarking Algorithm for Multiple Scan Chains in Design-for-Test. *Radioengineering* **2011**, *20*, 533–539.
20. Zhang, J.; Liu, L. Publicly Verifiable Watermarking for Intellectual Property Protection in FPGA Design. *IEEE Trans. VLSI Syst.* **2017**, *25*, 1520–1527. [CrossRef]
21. Sengupta, A.; Bhadauria, S. Untrusted Third-Party Digital IP Cores: Power-Delay Trade-off Driven Exploration of Hardware Trojan Secured Datapath during High-Level Synthesis. In Proceedings of the IEEE/ACM Great Lake Symposium on VLSI, Pittsburgh, PA, USA, 20–22 May 2015; pp. 167–172.
22. Cui, A.; Luo, Y.; Chang, C.H. Static and dynamic obfuscation of scan data against scan-based side-channel attacks. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 363–376. [CrossRef]
23. Kahng, A.B.; Lach, J.; Mangione-Smith, W.H.; Mantik, S.; Markov, I.L.; Potkonjak, M.; Tucker, P.; Wang, H.; Wolfe, J. Constraint-based watermarking techniques for design IP protection. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2001**, *20*, 1236–1252. [CrossRef]
24. Arunkumar, P.; Shangari, B. A New FSM Watermarking Method to Making Authorship Proof for Intellectual Property of Sequential Circuit Design Using STG. *IJMER* **2012**, *2*, 4159–4161.
25. Meana, R.W.P. Approximate Sub-Graph Isomorphism for Watermarking Finite State Machine Hardware. Master's Thesis, University of South Florida, Tampa, FL, USA, 2013.
26. Lofstrom, K.; Daasch, W.R.; Taylor, D. IC identification circuit using device mismatch. In Proceedings of the 2000 IEEE International Solid-State Circuits Conference, Digest of Technical Papers. San Francisco, CA, USA, 9 February 2002; pp. 372–373.
27. Bulens, P.; Standaert, F.X.; Quisquater, J.J. How to strongly link data and its medium: The paper case. *IET Inf. Secur.* **2010**, *4*, 125–136. [CrossRef]

28. Long, J.; Zhang, D.; Zuo, C.; Duan, J.; Huang, W. A robust low-overhead watermarking for field authentication of intellectual property cores. *Comput. Sci. Inf. Syst.* **2016**, *13*, 609–622. [CrossRef]
29. Xu, J.; Sheng, Y.; Liang, W.; Peng, L.; Long, J. A High Polymeric Mutual Mapping IP Watermarking Algorithm for FPGA Design. *J. Comput. Theor. Nanosci.* **2016**, *13*, 186–193. [CrossRef]
30. Cui, A.; Qu, G.; Zhang, Y. Ultra-Low Overhead Dynamic Watermarking on Scan Design for Hard IP Protection. *IEEE Trans. Inf. Forensics Secur.* **2017**, *10*, 2298–2313. [CrossRef]
31. Taha, A.; Hanbury, A. An Efficient Algorithm for Calculating the Exact Hausdorff Distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 2153–2163. [CrossRef] [PubMed]
32. Agarwal, K.; Fox, K.; Nath, A.; Sidiropoulos, A.; Wang, Y. Computing the Gromov-Hausdorff Distance for Metric Trees. *arXiv* **2015**, arXiv:1509.05751.