computers

MDPI

Article

L-PRNU: Low-Complexity Privacy-Preserving PRNU-Based Camera Attribution Scheme

Alan Huang and Justie Su-Tzu Juan *

Department of Computer Science & Information Engineering, National Chi Nan University, Puli, Nantou 54561, Taiwan; s108321007@mail1.ncnu.edu.tw
* Correspondence: jsjuan@ncnu.edu.tw

Abstract: A personal camera fingerprint can be created from images in social media by using Photo Response Non-Uniformity (PRNU) noise, which is used to identify whether an unknown picture belongs to them. Social media has become ubiquitous in recent years and many of us regularly share photos of our daily lives online. However, due to the ease of creating a PRNU-based camera fingerprint, the privacy leakage problem is taken more seriously. To address this issue, a security scheme based on Boneh–Goh–Nissim (BGN) encryption was proposed in 2021. While effective, the BGN encryption incurs a high run-time computational overhead due to its power computation. Therefore, we devised a new scheme to address this issue, employing polynomial encryption and pixel confusion methods, resulting in a computation time that is over ten times faster than BGN encryption. This eliminates the need to only send critical pixels to a Third-Party Expert in the previous method. Furthermore, our scheme does not require decryption, as polynomial encryption and pixel confusion do not alter the correlation value. Consequently, the scheme we presented surpasses previous methods in both theoretical analysis and experimental performance, being faster and more capable.

Keywords: low-complexity computation; PRNU-based camera attribution; secret sharing; privacy; camera fingerprint

Citation: Huang, A.; Juan, J.S.-T. L-PRNU: Low-Complexity Privacy-Preserving PRNU-Based Camera Attribution Scheme. *Computers* **2023**, *12*, 212. https://doi.org/10.3390/computers12100212

Academic Editors: Ali Sadiq, Houbing Song, Ahmad Fadhil Yusof, Sushil Kumar and Omprakash Kaiwartya

Received: 9 September 2023
Revised: 10 October 2023
Accepted: 18 October 2023
Published: 20 October 2023

1. Introduction

In recent years, the widespread adoption of social media, where many of us regularly share photos of our daily lives online, has introduced a significant cybersecurity concern. This issue revolves around the ease of security leakage related to Photo Response Non-Uniformity (PRNU) [1–5] noise, a digital fingerprint derived from the cameras used to capture these images. This personal camera fingerprint can be generated from images on social media and is employed to determine whether an unknown picture belongs to the individual. Due to the simplicity of creating a PRNU-based camera fingerprint, the issue of privacy leakage is now being viewed with heightened seriousness [6,7].

PRNU is a form of patterned noise that arises from imperfections during the manufacturing of Complementary Metal–Oxide-Semiconductor (CMOS) sensors [8,9]. When CMOS sensors are created, the presence of contaminants affects how responsive the sensor is to light, ultimately resulting in lower-quality images in cameras. What is unique is that each camera's CMOS sensor generates its own distinct PRNU noise pattern. By utilizing the PRNU noise identification technique, individuals can easily determine the owner of an anonymous image. This noise-based identification method, for example, can be used to verify the source of evidence related to illegal activities such as child pornography. Below, we provide more details on this case:

A notable celebrity is involved in a high-profile child pornography case. During the trial, the celebrity's name should be kept confidential. Somehow, when matching the suspect's camera fingerprint with child pornography images, the third-party expert

accidentally leaks the suspect's camera fingerprint to a journalist. Due to the widespread pictures that was uploaded online, the journalist identified the suspect by comparing it to camera fingerprints generated by every possible high-profile celebrity. The celebrity's name was then published in major news, damaging his reputation and career, even though he was later proven innocent. This incident highlights the importance of researching a safe PRNU-based camera attribution scheme.

Although these cases may be difficult to believe, the absence of protection for the judgment scheme based on PRNU camera fingerprints remains a significant concern. In January 2021, a security solution (e-PRNU) applying Boneh–Goh–Nissim (BGN) encryption was introduced to address this issue [10]. However, BGN encryption comes with a substantial computational overhead due to its power computation. In June 2021, another approach called SSS-PRNU, based on Shamir Secret Sharing [11,12], was presented with a lower computational overhead. Nevertheless, the SSS-PRNU scheme [13] suffers from a privacy leakage problem, which compromises its effectiveness.

In order to resolve the shortcomings of the previous schemes, we develop a low-complexity privacy-preserving PRNU-based camera attribution scheme, called the L-PRNU scheme, that utilizes polynomial encryption and pixel confusion. This new scheme offers a computation time that is more than ten times faster than the e-PRNU scheme and avoids any privacy leakage concerns. In L-PRNU, while keeping the previous scheme's advantage, we made some improvements to overcome the problem that earlier models met. The following are the key contributions we made:

1. Low computational overhead;

Switching from the BGN encryption method to polynomial encryption in the L-PRNU scheme significantly reduces the time spent on encryption compared to the e-PRNU scheme.

2. No need for decryption;

Due to the use of a polynomial equation as the encryption method in L-PRNU, the true correlation number, which determines the picture's ownership, can be calculated without the necessity of decryption. Saves huge computational power and time compared to the previous scheme.

3. Privacy-preserving.

The scheme combines privacy techniques like pixel shuffling and polynomial encryption to protect sensitive data. These methods deter collusion among experts and make it difficult to reveal the true correlation value or reconstruct the camera fingerprint. The data are transformed into an indecipherable format while transferring. Pixel shuffling further obscures the images, making unauthorized access nearly impossible. These safeguards protect camera owners' identities and prevent unauthorized data disclosure.

In addition to the three main points mentioned earlier, we offer flexibility in the selection of third-party experts. Ideally, you should choose three different organizations or three separate environments for enhanced security. Increasing the number of third-party experts can further enhance security in the scenario. Additionally, a higher number of third-party experts can lead to reduced computational time.

The remainder of this paper is structured as follows. Section 2 presents the background of the PRNU-based source camera attribution method, and the related work—e-PRNU scheme, and Section 3 describes the architecture of the proposed scheme. The results of the experiment and the analysis of different schemes' computations are given in Sections 4 and 5. Finally, Section 6 presents the conclusion. Further information will be presented after Section 6.

2. Background and Related Work

This section presents some important research related to the main method of this paper.

2.1. Background

PRNU-Based Source Camera Attribution [1–5]

During the manufacturing process of camera sensors, minor variations can result in Photo Response Non-Uniformity (PRNU), which is a unique pattern noise specific to each camera. This characteristic allows for the extraction and identification of PRNU-based noise from images taken by different cameras. The camera image output I can be represented as follows:

$$I = I_0 + I_0 X + \varphi \tag{1}$$

Here, I_0 represents the noise-free image, X denotes the PRNU noise, and φ represents the random noise. By utilizing a wavelet denoising filter f [14], we can easily obtain the noise-free image I_0. To mitigate the impact of random noise φ on the camera fingerprint, the camera fingerprint F can be estimated by combining the N extracted PRNU noise images using the formula $F = \frac{\sum(I - f(I))}{N}$. To determine the source of an unknown image I', we can extract a PRNU noise F' using the aforementioned algorithm: $F' = I' - f(I')$. Subsequently, we take the camera fingerprint F by calculating the correlation coefficient of F' and F, where it is possible to ascertain whether the owner of I' matches the owner of the images in the collection.

$$r(F,\ F') = \frac{\sum_{i=1}^{n}\left(F_i - \overline{F}\right)\left(F_i' - \overline{F'}\right)}{\sqrt{\sum_{i=1}^{n}\left(F_i - \overline{F}\right)^2}\sqrt{\sum_{i=1}^{n}\left(F_i' - \overline{F'}\right)^2}} \tag{2}$$

where n is the total pixels of the noise image F' (=the total pixels of the fingerprint F), F_i is a pixel in F, F_i' is the corresponding pixel in F', $\overline{F'} = \sum F_i'/n$ is the average pixel value of F', and $\overline{F} = \sum F_i/n$ is the average pixel value of the camera fingerprint F. If $r(F, F')$ exceeds a specific threshold, which confirms that both F and F' originated from the same camera sensor. The threshold value may vary depending on the specific camera sensor being used.

2.2. Related Work

e-PRNU [10,15]

In 2021, Mohanty et al. introduced a privacy-preserving approach based on the Boneh–Goh–Nissim (BGN) encryption scheme. With the majority of camera attribution tasks outsourced to a third-party entity by law enforcement agencies, the need for complex or computationally intensive tasks is eliminated. The system is divided into five entities, as illustrated in Figure 1. This system is also referenced by the other literatures, such as Jena et al. who introduced SSS-PRNU [13].

1. Fingerprint Source: This entity, an organization affiliated with law enforcement, extracts camera fingerprints from a set of known images. It is assumed to be trustworthy and secure. To link the fingerprints with online photos, the fingerprints are encrypted using BGN encryption and transmitted to the Third-Party Expert fingerprint storage. The encryption key is provided by the KMA;
2. Third-Party Expert: This entity stores all the encrypted fingerprints received from the Fingerprint Source. Privacy leakage is a concern, so this entity should not have knowledge of the fingerprint owners. The Third-Party Expert requires ample storage space and sufficient computing power to handle high-computation tasks. According to the paper, this entity can be a public cloud service like Amazon or Google;
3. Match Maker: Representing a trusted organization such as a law enforcement authority or judge, the Match Maker aims to match a given picture with a known set of pictures. While the Fingerprint Source manages the known set of pictures, the Match Maker extracts the noise from the picture and encrypts the fingerprint using BGN encryption, similar to the Fingerprint Source. Ultimately, the Match Maker or judge will make a judgment using the threshold and the value provided by the Match Maker Server;

4. Match Maker Server: This entity operates under the control of the Match Maker and is considered secure. Its role is to decrypt the first part correlation sent by the Third-Party Expert. Additionally, the Match Maker Server performs the second part of the correlation to obtain the actual correlation value. The Match Maker Server requires some computational ability for decryption but does not need to solve complex problems;
5. KMA: This trusted entity is responsible for generating the public and private keys CID, CID′ used in BGN encryption.

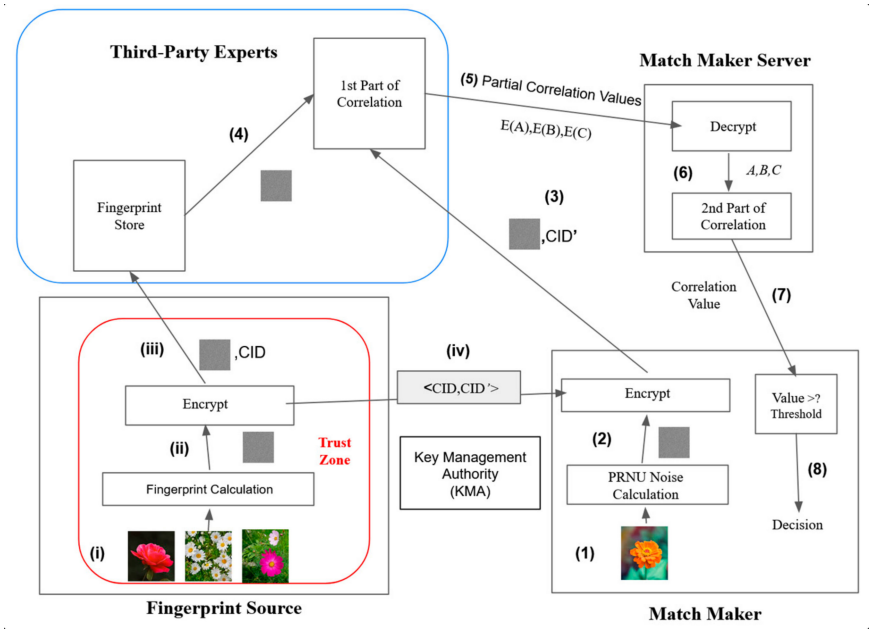

Figure 1. The Methodology of e-PRNU.

Since BGN encryption is homomorphic with respect to additions and multiplications, the correlation in the PRNU-Based Source camera attribution can be divided into two parts without affecting the final correlation value. The correlation is split as follows:

First part correlation: This involves collecting two encrypted fingerprints, one known $E(F)$ and the other unknown $E(F')$. The Third-Party Expert calculates three partial correlation values: $E(A)$, $E(B)$, and $E(C)$ using the following equations:

$$E(A) = \sum_{i=1}^{n} \left(E(F_i) - E(\overline{F})\right)\left(E(F_i') - E\left(\overline{F'}\right)\right), \tag{3}$$

$$E(B) = \sum_{i=1}^{n} \left(E(F_i) - E(\overline{F})\right)^2, \tag{4}$$

$$E(C) = \sum_{i=1}^{n} \left(E(F_i') - E\left(\overline{F}'\right)\right)^2. \tag{5}$$

The Third-Party Expert sends the three encrypted partial correlation values $E(A)$, $E(B)$, and $E(C)$ to the Match Maker Server.

Second part correlation: The Third-Party Expert receives the three encrypted partial correlation values ($E(A)$, $E(B)$, and $E(C)$) and decrypts them to obtain A, B, and C, for $A = \sum_{i=1}^{n} \left(F_i - \overline{F}\right)\left(F_i' - \overline{F'}\right)$, $B = \sum_{i=1}^{n} \left(F_i - \overline{F}\right)^2$, and $C = \sum_{i=1}^{n} \left(F_i' - \overline{F'}\right)^2$. Utilizing the homomorphic property of BGN encryption, dividing A, B, and C enables the derivation of the actual correlation value.

$$r(F, F') = \frac{A}{\sqrt{BC}} \tag{6}$$

3. Materials and Methods

3.1. Main Scheme

L-PRNU is a system that calculates the camera fingerprint using a known query and an unknown source image. By matching these two crucial elements within the L-PRNU system, privacy breaches are prevented. The following section will provide a detailed explanation of the L-PRNU scheme, as illustrated in Figure 2.

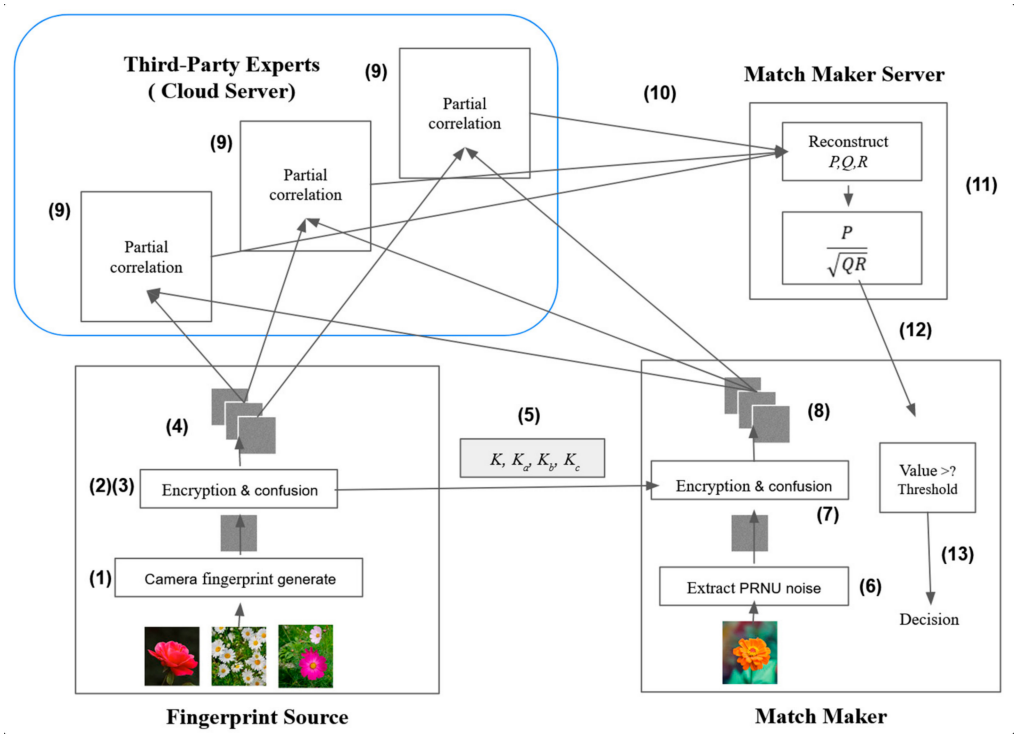

Figure 2. The Methodology of the Proposed L-PRNU.

3.1.1. System Model

1. Fingerprint Source: This entity is considered to be a trustable entity. It is responsible to create a key K and (at least) three disorder keys, K_a, K_b, and K_c, so that the Match Maker can encrypt the unknown photo in the same way. While the Match Maker extracts the PRNU noise from the unknown photo, the Fingerprint makes the fingerprint out of the known set of photos. In addition, it encrypts the fingerprint into three encrypted photos.
2. Third-Party Expert: The Third-Party Expert consists of multiple organizations or cloud services with powerful computational capabilities, but it is not inherently trusted. Its task involves generating partial correlations from 2×3 encrypted photos received from the Fingerprint Source and the Match Maker. Once the partial correlation numbers are calculated, they are sent to the Match Maker Server. This role can be assigned to different organizations or a single organization.
3. Match Maker Server: The Match Maker Server and the Match Maker both operate within the jurisdiction of law enforcement authorities. While the Third-Party Expert handles most of the computation, the Match Maker Server combines the 21 partial correlation numbers with the final correlation. Due to the relatively low complexity of the encryption, solving this problem requires only occasional computational power. Once the combination process is completed, the Match Maker Server provides the final correlation back to the Match Maker.
4. Match Maker: This entity represents the law enforcement authority or the judge. The Match Maker uses the key K and three disorder keys, K_a, K_b, and K_c, provided by the Fingerprint Source to encrypt the unknown evidence into three encrypted photos. At

the conclusion of the entire scheme, the Match Maker combines the final correlation with the camera's threshold to make the ultimate judgment.

3.1.2. The Process

- Fingerprint Source:
  1. A set of known images that came from the same camera will be collected. All the images will be denoised and extracted into a camera fingerprint F.
  2. Fingerprint Source uses the key K to encrypt the camera fingerprint into three encrypted photos A, B, and C.
  3. The encrypted photos will be confused via three disorder keys K_a, K_b, and K_c. Making $K_a(A)$, $K_a(B)$, $K_b(B)$, $K_b(C)$, $K_c(C)$, and $K_c(A)$. Obtaining six photos that are encrypted and disarranged.
  4. Distribute the encrypted photos to the Third-Party Expert according to the Division algorithm.
  5. Fingerprint Source passes the key K and three disorder keys K_a, K_b, and K_c to Match Make via a secret channel.
- Match Maker:
  6. Extract the PRNU noise picture F' from an unknown evidence image and encrypt it with K into three encrypt photos A', B', and C'.
  7. Disarrange the three encrypted PRNU noise pictures into six confusion images with three disorder keys K_a, K_b, and K_c: $K_a(A')$, $K_a(B')$, $K_b(B')$, $K_b(C')$, $K_c(C')$, and $K_c(A')$.
  8. Distribute the encrypted images to the Third-Party Expert according to the Division algorithm.
- Third-Party Expert:
  9. Each of the nodes of the Third-Party Expert will receive the encrypted images that their task assigns. While some are from the Fingerprint Source, the others are from the Match Maker. Each node will receive the mission that they are assigned, noting that they will not know which images (A, or B, or C) they received.
  10. Occurs after each node finishes making the partial correlation number according to equation (8). They will send the numbers to the Match Maker Server.
- Match Maker Server:
  11. After receiving all the partial correlation numbers, the Match Maker will combine them into the real correlation coefficient with the combine algorithm.
  12. The Match Maker Server will send the real correlation coefficient back to the Match Maker
- Match Maker:
  13. The final decision is made by the Match Maker based on the final correlation value received from the Match Maker server. If the correlation value is equal to or above the camera sensor's PRNU threshold, the unknown image is proved to be taken from the suspected camera or else not.

3.2. Solution Details

Supposing there is a scenario where we need to determine whether an unidentified photo belongs to a specific individual. The PRNU-Based Source camera attribution method requires a collection of images provided by that individual. Initially, the Fingerprint Source creates a camera fingerprint F using the set of provided images. Then, the Fingerprint Source generates a value K to encrypt the camera fingerprint F. Multiplying K with each pixel p and dividing the result by $d = 256$ twice yields a polynomial equation as follows:

$$p \times K = (a \times d + b) \times d + c \tag{7}$$

Here, p represents each pixel in the camera fingerprint F, K is a randomly generated key used to encrypt each pixel within the range of 2 to $2^{16} - 1$, and d is the value by which we intend to divide. To ensure that a, b, and c remain within the range of 0 to 255 and can represent an image, d is set to 256 (However, if we are willing to transmit a numerical array instead of an image, d can be set to a larger value.) By converting each pixel p in the camera fingerprint F into a polynomial, we obtain three encrypted images A, B, and C formed by a, b, and c, respectively.

Assuming the presence of three third-party experts, before transmitting these three images to each expert, the Fingerprint Source confuses each pair of the encrypted image ((A, B), (B, C), and (A, C)) with a disorder key (If the number of third-party experts are more than three, the number of disorder keys can be more than three.) In this case, (A, B) is confused with K_a, (B, C), with K_b, and (C, A) with K_c. Using the disorder algorithm, Third-Party Experts can calculate partial values, but they cannot reconstruct the original camera fingerprint by colluding with other experts or guessing the true value of F. There is only one Third-Party Expert.

Next, the division algorithm is employed to send the three confused images to the Third-Party Experts. Simultaneously, the Match Maker extracts the PRNU noise F' from the unknown photo. Using the same encryption key K with (7) provided by the Fingerprint Source, the PRNU noise F' is encrypted into three encrypted images A', B', and C'. Subsequently, these three encrypted images are also encrypted via the disorder keys K_a, K_b, and K_c and the resulting pairs will be transmitted to the Third-Party Experts using the division algorithm.

Note that, although the Fingerprint Source shuffles the positions of the pixels, because the Match Maker shuffles them in the same manner, the true correlation coefficient remains unchanged.

Each Third-Party Expert receives a set of up to four encrypted images. They need to compute all combinations of each pair of encrypted images, including computing with itself. For example, if one receives $K_a(A)$, $K_a(A')$, $K_a(B)$, and $K_a(B')$ as the images, the expert has to calculate the combinations $Pr(K_a(A),\ K_a(A))$, $Pr(K_a(A),\ K_a(A'))$, $Pr(K_a(A'),\ K_a(A'))$, $Pr(K_a(A),\ K_a(B))$, $Pr(K_a(A'),\ K_a(B))$, $Pr(K_a(A),\ K_a(B'))$, and $Pr(K_a(A'),\ K_a(B'))$. The partial correlation function is represented as follows, where α_i (β_i, respectively) means each pixel on α (β, respectively):

$$Pr(\alpha, \beta) = \sum_{i=1}^{n} (\alpha_i - \overline{\alpha}) \times \left(\beta_i - \overline{\beta}\right) \tag{8}$$

Once the Third-Party Experts complete their calculations, they send the partial correlation numbers to the Match Maker server. The Match Maker server collects all the partial correlation numbers and combines them using a combine algorithm to obtain P, Q, and R. Having P, Q, and R allows the Match Maker server to reconstruct the true correlation coefficient using the equation $\frac{P}{\sqrt{QR}}$. Finally, the Match Maker server sends the real correlation coefficient back to the Match Maker. The Match Maker then determines whether the unknown photo belongs to the specific individual based on a camera sensor threshold. If the correlation coefficient exceeds the threshold, it indicates a match with the same camera; otherwise, it does not.

3.2.1. Algorithm 1: Disorder Algorithm

To begin, the Fingerprint Source generates a disorder key—a permutation function—whose domain comprises all pixel positions of the input photo. This key rearranges the pixels in the input photo, effectively transforming it into a confusion photo. That is, each pixel is relocated to a random position within the photo. For instance, $K_a(A)$ will transform photo A into a confusion photo.

When this same disorder key K_a is sent to the Match Maker server, it can rearrange another photo A' using the same method to be $K_a(A')$. Two pixels located at the same position of A and A' will be respectively arranged to another same position in $K_a(A)$ and $K_a(A')$. Users can decide the permutation function by themselves. The following is an

example of a permutation function for the disorder algorithm. Where the position range of all pixels in one $m \times n$ image is (0, 0) to $(m-1, n-1)$, and gcd (a, b) represents the greatest common divisor of a and b.

Algorithm 1 Disorder algorithm

Input: an image A with size $m \times n$, the permutation key K with gcd $(K, mn) = 1$.
Output: the rearrange image $K(A)$.
Step1. For any pixel (i, j) of A, do Step 2–3.
Step2. Let $Q = K(n(i-1) + j) \bmod mn$.
Step3. Let the value of pixel $(\lfloor Q/n \rfloor, Q \bmod n)$ of $K(A)$ be the value of pixel (i, j) of A.

Note that the usage of the permutation function guarantees that the partial correlation remains unaffected by this disorder process.

3.2.2. Algorithm 2: Division Algorithm

Continuing from the encryption process described in (7), the original camera fingerprint image Y is transformed into three images: A, B, and C. To fully grasp the entire procedure, we need to refer back to the encryption algorithm outlined in (7). Each pixel in the original image is multiplied by a factor of K. By substituting F, F' with KF, KF', we can compute the new correlation using the following equation:

$$r(KF,\ KF') = \frac{K^2 P}{\sqrt{K^2 Q \times K^2 R}} = \frac{P}{\sqrt{QR}} = r(F,\ F') \tag{9}$$

Since all of P, Q, and R are multiplied by the same value K, the correlation remains unchanged after the encryption process described in (7). Similarly, by substituting F and F' with $A \times d^2 + B \times d + C$ and $A' \times d^2 + B' \times d + C'$, respectively, we can observe that the correlation coefficient is composed of smaller computations involving the difference between pixels and their average values. This leads us to the partial correlation function $Pr(\alpha, \beta)$ as Formula (8). The complete correlation coefficient is then split into 21 numbers $r_i = Pr(\alpha, \beta)$, as Table 1 shows, where $\gamma = a$, b, or c according to the color of the background. For example, $r_{14} = Pr(K_c(A), K_c(C)) = Pr(K_c(C), K_c(A))$.

Table 1. Values and Indexes of Partial Correlation Function $Pr(\alpha, \beta)$.

α \ β	$K_\gamma\ (A)$	$K_\gamma\ (B)$	$K_\gamma\ (C)$	$K_\gamma\ (A')$	$K_\gamma\ (B')$	$K_\gamma\ (C')$
$K_\gamma\ (A)$	r_{10}	r_{12}	r_{14}	r_1	r_2	r_3
$K_\gamma\ (B)$	r_{12}	r_{11}	r_{13}	r_4	r_5	r_6
$K_\gamma\ (C)$	r_{14}	r_{13}	r_{15}	r_7	r_8	r_9
$K_\gamma\ (A')$	r_1	r_4	r_7	r_{16}	r_{18}	r_{20}
$K_\gamma\ (B')$	r_2	r_5	r_8	r_{18}	r_{17}	r_{19}
$K_\gamma\ (C')$	r_3	r_6	r_9	r_{20}	r_{19}	r_{21}

These 21 partial correlation numbers will be assigned to the three Three-Party Experts to complete evenly. To ensure privacy, the first Third-Party Expert receives images $K_a(A)$, $K_a(A')$, $K_a(B)$, $K_a(B')$ that have been confused via the disorder key K_a. The images received by the second Third-Party Expert have been confused via the disorder key K_b, for the images $K_b(B)$, $K_b(B')$, $K_b(C)$, $K_b(C')$; while the last Third-Party Expert receives the confused images $K_c(A)$, $K_c(A')$, $K_c(C)$, $K_c(C')$ that have been confused via the disorder key K_c. Each Third-Party Expert is not provided with combinations of A, B, and C or A', B', and C', and each image is confused with a distinct disorder key. As in Table 1, assigning the partial correlations with the same color of the background to the same expert is one of the assignment methods. The process of the Division algorithm for the fingerprint source

and match maker should be as follows ((4) and (8) in Figure 2) when there are three Three-Party Experts:

Algorithm 2 Division

Step1. The entity that receives three images A, B, C that are encrypted.
Step2. The entity computes $K_a(A)$, $K_c(A)$, $K_a(A')$, $K_c(A')$, $K_a(B)$, $K_b(B)$, $K_a(B')$, $K_b(B')$, $K_b(C)$, $K_c(C)$, $K_b(C')$, $K_c(C')$ via Algorithm 1 with disorder keys K_a, K_b, and K_c.
Step3. The entity sends images $K_a(A)$, $K_a(A')$, $K_a(B)$, $K_a(B')$ to the first Third-Party Expert; $K_b(B)$, $K_b(B')$, $K_b(C)$, $K_b(C')$ to the second Third-Party Expert; $K_c(A)$, $K_c(A')$, $K_c(C)$, $K_c(C')$ to the third Third-Party Expert.

The experts will only receive images with ids, of which they will not know which part of image they are calculating. After the computation process $Pr(\alpha, \beta)$ as Formula (8), the experts will send the partial numbers back to the Match Maker Server to end their mission.

3.2.3. Algorithm 3: Combine Algorithm

After obtaining the 21 partial correlation numbers, according to Formulas (7) and (8), Table 1, and the definitions of P, Q, R, the value of P, Q, R can be restored via the equations below:

$$P = d^4 \times r_1 + d^3 \times r_2 + d^2 \times r_3 + d^3 \times r_4 + d^2 \times r_5 + d \times r_6 + d^2 \times r_7 + d \times r_8 + r_9 \quad (10)$$

$$Q = d^4 \times r_{10} + d^2 \times r_{11} + 2 \times d^3 \times r_{12} + 2 \times d \times r_{13} + 2 \times d^2 \times r_{14} + r_{15} \quad (11)$$

$$R = d^4 \times r_{16} + d^2 \times r_{17} + 2 \times d^3 \times r_{18} + 2 \times d \times r_{19} + 2 \times d^2 \times r_{20} + r_{21} \quad (12)$$

The Match Maker Server will receive 21 numbers of partial numbers that were calculated from the third-party experts. Using Formula (10)–(12), the Match Maker Server will reconstruct P, Q, R. Finally, the Match Maker Server obtains the final correlation number via Formula (9). The following are the pseudo-code of the algorithm:

Algorithm 3 Combine algorithm

Step1. Collect all the partial numbers r_1–r_{21} from the third-party expert.
Step2. Calculate P, Q, and R with Formula (10)–(12).
Step3. Reconstruct the origin correlation number $r(F,\ F') = \frac{P}{\sqrt{QR}}$.

4. Results

It is crucial to take into account both the performance and security aspects of the L-PRNU scheme. In this section, we present an analysis that goes beyond these fundamental components.

4.1. Computation Analysis

In order to assess the performance of the L-PRNU scheme, we deployed a specific configuration consisting of one node for the Fingerprint Source, three nodes for the Third-Party Expert, one node for the Match Maker, and one node for the Match Maker Server. Each node was equipped with an Intel(R) Xeon(R) CPU featuring two cores clocked at 2.20 GHz and 12 GB RAM. To replicate real-world conditions, all entities were isolated in separate containers, simulating the actual environment. The L-PRNU code was implemented using Python 3.10 and executed on the Google Colab platform. For testing purposes, an image with a resolution of 720 × 720 pixels was utilized.

Regarding the computation complexity within the L-PRNU scheme, the encryption algorithm (division and disorder) employed by the Fingerprint Source and the Match Maker entailed one multiplication, two divisions, and three permutation functions that disordered

the encrypted photos. Since decryption was not necessary, the decryption computation complexity was negligible. In the case of the Third-Party Expert, the complexity of the partial correlation calculation involved two subtractions, one multiplication per pixel, and a $n - 1$ (n represents the total pixel of a photo) addition, plus one division to calculate the total pixel average of F and F'. Lastly, the combined algorithm implemented within the Match Maker Server comprised a total of 19 multiplications, one division, one square root, and 18 additions.

When we compare the computational complexity to that of the e-PRNU scheme, it becomes apparent that the encryption algorithm in the L-PRNU scheme is relatively simpler. In the e-PRNU encryption process, there is a single round-off operation, three exponentiations, and two multiplications for each pixel. In contrast, the decryption algorithm in the e-PRNU scheme is considerably more computationally demanding, necessitating 12 exponentiations, three discrete logarithmic calculations, and one division operation for each pixel. This computational intensity in the e-PRNU scheme has led to the adoption of a strategy in which only the critical pixels are processed, rather than the entire image, in order to manage the substantial computational time involved.

To evaluate the performance of the L-PRNU scheme, we conducted comparative tests with the e-PRNU scheme using different picture sizes (see Table 2). In the case of a 100×100 resolution, L-PRNU showcased considerably faster testing times, ranging from 4.448 s in a single node of the Third-Party Expert to 1.482 s in three nodes. On the contrary, e-PRNU exhibited much longer testing times, with the highest recorded time being 404.17 s in a single node of the Third-Party Expert and 102.58 s in four nodes. When it came to a 720×720 resolution, L-PRNU once again outperformed e-PRNU, with testing times ranging from 21.903 s in three nodes of the Third-Party Expert to 65.709 s in a single node. In contrast, e-PRNU took an excessive amount of time to complete the test, indicating an impractical performance.

Table 2. The Comparison of Computation time in The Third-Party Expert entity between L-PRNU and e-PRNU.

Picture Size	Testing Scheme	
	L-PRNU	**e-PRNU**
100×100	4.448 s (#1 node) 1.482 s (#3 nodes) 0.847 s (#6 nodes)	404.17 s (#1 node) 102.58 s (#4 nodes) 26.60 s (#16 nodes)
720×720	65.709 s (#1 node) 21.903 s (#3 nodes) 12.516 s (#6 nodes)	>11 min

In addition, the L-PRNU scheme offers the advantage of not requiring decryption and utilizes polynomial encryption. As a result, both the Match Maker Server and Match Maker entities exhibit faster computation times when compared to the e-PRNU scheme (see Table 3).

Table 3. The Comparison of Computation Time of the Other Entities between L-PRNU and e-PRNU.

Entity	Computation Tasks	L-PRNU	e-PRNU
Match Maker server	Decrypt and Second Part of Correlation	0.012 s	23.05 s
Match Maker	Encrypt and PRNU Noise Calculation	6.986 s	42.73 s

In summary, the performance analysis indicates that the L-PRNU scheme relies on basic mathematical operations, avoiding complex exponential calculations like e-PRNU. Consequently, L-PRNU achieves significantly lower computation times, making it a more efficient solution for PRNU-based camera attribution tasks.

4.2. Security Analysis

In this subsection, we provide a thorough examination of the security aspects of the L-PRNU scheme from various perspectives. These security considerations are discussed in detail below.

1. Privacy Preservation: The L-PRNU scheme employs multiple techniques, such as the shuffling of pixel positions and polynomial encryption, to ensure the preservation of privacy. These measures prevent collusion among third-party experts and make it extremely challenging to infer the true correlation value or reconstruct the original camera fingerprint. By encrypting the camera fingerprint and PRNU noise images using polynomial encryption, the sensitive data is transformed into an unintelligible form that can only be deciphered with the specific decryption key K (ranging from 2 to $2^{16} - 1$), which is difficult to guess. Additionally, the scheme introduces shuffling to further obfuscate the images, making it virtually impossible for unauthorized parties to gain access to the original content. These privacy-preserving techniques are crucial in safeguarding the identities of camera owners and preventing the unauthorized disclosure of their personal information.
2. Homomorphic Property: The scheme leverages the homomorphic property of encryption, specifically in the BGN encryption used in previous schemes. The homomorphic property allows for dividing the correlation calculation into some parts without affecting the final correlation value. This property ensures that the encryption does not compromise the integrity or accuracy of the correlation calculations while maintaining privacy.
3. Different Disorder Keys: The scheme utilizes disorder keys to further enhance security. The encryption images are confused with distinct disorder keys, making it difficult for Third-Party Experts to correlate the encrypted images and reveal sensitive information about the camera fingerprint or PRNU noise, even if all Third-Party Experts colluded. This additional layer of pixel confusion adds complexity to potential attacks and enhances the security of the scheme.
4. Secure Entities: The scheme defines various entities involved in the process, such as the Fingerprint Source, Third-Party Experts, Match Maker, and Match Maker Server. These entities have specific roles and responsibilities in the scheme, ensuring that the sensitive data is handled by trusted and secure parties. The proper management and authentication of these entities are essential for maintaining the overall security of the scheme.
5. Highly Secure Key Management: In the L-PRNU scheme, all the keys (K, K_a, K_b, K_c) are exclusively transferred between trusted entities (Fingerprint Source and Match Maker). Furthermore, the scheme does not necessitate decryption, eliminating any potential risks associated with key leakage during the decryption process. This ensures the confidentiality and integrity of the keys, further bolstering the overall security of the scheme.

Overall, the L-PRNU scheme presents a comprehensive and effective approach to address the security and privacy concerns associated with PRNU-based camera attribution. Via the integration of encryption, pixel confusion, secure entities, and a strong focus on key management, the scheme establishes a robust security framework.

5. Discussion

In this paper, we have tackled the pressing concern of privacy breaches related to the creation of personal camera fingerprints derived from images shared on social media. Initially, we discuss the introduction of the e-PRNU security solution in January 2021, which

made use of Boneh–Goh–Nissim (BGN) encryption. Although this approach was effective, it suffered from a substantial computational overhead. In response to this challenge, we introduce our groundbreaking L-PRNU scheme, which utilizes polynomial encryption and pixel obfuscation techniques. This novel scheme not only achieves computation times more than ten times faster than e-PRNU, but it also eliminates the need for decryption, thereby enhancing its practicality. Furthermore, our approach places a strong emphasis on preserving privacy by incorporating methods such as pixel shuffling and polynomial encryption, rendering it exceedingly difficult to discern the true correlation value or reconstruct the camera fingerprint. This scheme presents several notable advantages compared to previous methodologies.

- First and foremost, our innovative approach achieves a remarkable computation time reduction, exceeding ten times the speed of the existing e-PRNU scheme. With our polynomial encryption method, it significant accelerates the previously burdensome runtime computational overhead associated with BGN encryption in e-PRNU.
- Moreover, compared with the BGN decryption in the e-PRNU scheme, our scheme dispenses the need for decryption, thanks to the application of polynomial encryption and pixel disorder, which leave the correlation value unaltered. This breakthrough obviates the necessity to transmit critical pixels to a third party, a requirement of previous methods. By maintaining the original image format during transmission, we uphold the data integrity and preserve the visual information.
- Furthermore, as we enhance the computation speed, we continue to prioritize privacy preservation. Our approach places significant importance on safeguarding privacy by implementing pixel position shuffling. This not only discourages collaboration among external experts but also significantly hinders any efforts to determine the actual correlation value or reconstruct the original camera fingerprint. These crucial contributions collectively bolster the security and effectiveness of our L-PRNU scheme in addressing the privacy issues linked to personal camera fingerprints obtained from social media images.

In order to clearly see the advantages of the proposed scheme, Figure 3 shows the calculation time of the third-party expert entity between L-PRNU and e-PRNU in Table 2 for 100×100 images. It is easy to see from this figure that the time saved by the proposed scheme is close to a hundred times, which makes the proposed scheme more practical than the previous method.

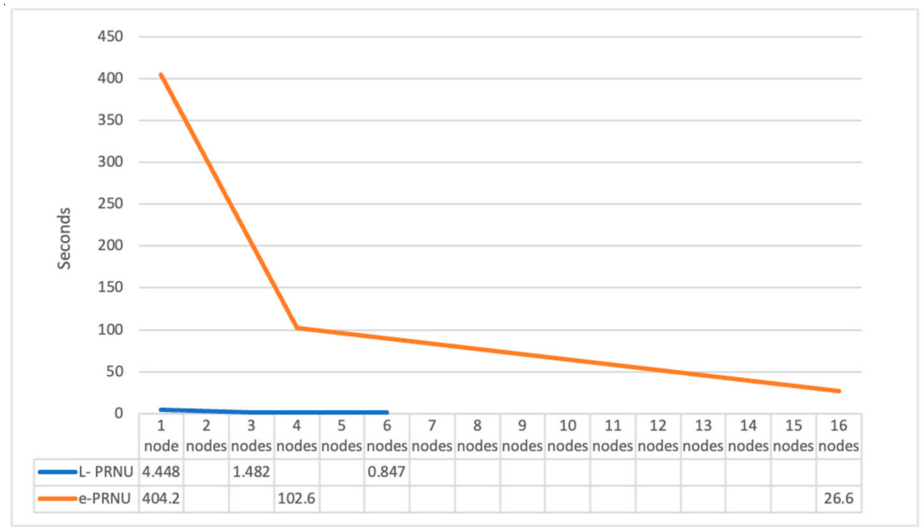

Figure 3. The Computation time in The Third-Party Expert entity between L-PRNU and e-PRNU for 100×100 image.

6. Conclusions

In conclusion, our comprehensive analysis unequivocally highlights the exceptional performance and security features of the L-PRNU scheme. The depth of our evaluation goes beyond the fundamental components, providing a nuanced understanding of its strengths. In the realm of performance, the L-PRNU scheme exhibits a remarkable advantage over its predecessor, e-PRNU. With significantly reduced computation times exceeding tenfold in some cases, it effectively eliminates the burdensome computational overhead associated with previous encryption methods. Moreover, its ingenious use of polynomial encryption and pixel obfuscation obviates the need for decryption, preserving the data integrity and streamlining the process. From a security perspective, the L-PRNU scheme employs a multifaceted approach that fortifies privacy preservation. The integration of pixel shuffling, polynomial encryption, and the use of distinct disorder keys creates formidable barriers against collusion and unauthorized access to sensitive data. The scheme's secure entity framework and meticulous key management further enhance its security, ensuring that trusted parties handle critical information.

Our scheme offers a comprehensive, efficient, and highly secure solution for PRNU-based camera attribution, effectively mitigating privacy risks associated with social media images. While our current security measures are robust, we acknowledge the ever-evolving technological landscape and the potential for emerging privacy vulnerabilities. Therefore, future research should focus on continuous improvements to encryption methods and practical implementations, ensuring the ongoing protection of individual privacy in the digital age. The L-PRNU scheme stands as a significant advancement in the field of camera attribution, addressing both the performance and security concerns with unparalleled effectiveness and sophistication.

Author Contributions: Conceptualization, J.S.-T.J. and A.H.; methodology, J.S.-T.J.; software, A.H.; validation, J.S.-T.J.; formal analysis, J.S.-T.J. and A.H.; investigation, A.H.; data curation, A.H.; writing—original draft preparation, A.H.; writing—review and editing, J.S.-T.J.; visualization, A.H.; supervision, J.S.-T.J.; project administration, J.S.-T.J.; funding acquisition, J.S.-T.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the National Science and Technology Council, R.O.C. under grant NSTC 111-2115-M-260 -001 - and NSTC 112-2115-M-260 -001 -MY2.

Data Availability Statement: The datasets generated analyzed during this current study are not publicly available due to participant privacy, but they are available from the corresponding author on reasonable request.

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

References

1. Bartlow, N.; Kalka, N.; Cukic, B.; Ross, A. Identifying sensors from fingerprint images. In Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Miami, FL, USA, 20–25 June 2009; IEEE: Piscataway, NJ, USA, 2009.
2. Dirik, A.E.; Sencar, H.T.; Memon, N. Analysis of seam-carving-based anonymization of images against PRNU noise pattern-based source attribution. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 2277–2290. [CrossRef]
3. Lukáš, J.; Fridrich, J.; Goljan, M. Detecting digital image forgeries using sensor pattern noise. In *Security, Steganography, and Watermarking of Multimedia Contents VIII*; SPIE: Washington, DC, USA, 2006; Volume 6072, pp. 362–372.
4. Lukas, J.; Fridrich, J.; Goljan, M. Digital camera identification from sensor pattern noise. *IEEE Trans. Inf. Forensics Secur.* **2006**, *1*, 205–214. [CrossRef]
5. Taspinar, S.; Mohanty, M.; Memon, N. PRNU-based camera attribution from multiple seam-carved images. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 3065–3080. [CrossRef]
6. Bertini, F.; Sharma, R.; Iannì, A.; Montesi, D. Smartphone verification and user profiles linking across social networks by camera fingerprinting. In Proceedings of the Digital Forensics and Cyber Crime: 7th International Conference, ICDF2C 2015, Seoul, Republic of Korea, 6–8 October 2015; Revised Selected Papers 7. Springer International Publishing: Berlin/Heidelberg, Germany, 2015.
7. Fernández-Menduina, S.; Pérez-González, F. On the information leakage of camera fingerprint estimates. *arXiv* **2020**, arXiv:2002.11162.

8. Dirik, A.E.; Sencar, H.T.; Memon, N. Digital single lens reflex camera identification from traces of sensor dust. *IEEE Trans. Inf. Forensics Secur.* **2008**, *3*, 539–552. [CrossRef]
9. Dirik, A.E.; Karaküçük, A. Forensic use of photo response non-uniformity of imaging sensors and a counter method. *Opt. Express* **2014**, *22*, 470–482. [CrossRef] [PubMed]
10. Mohanty, M.; Zhang, M.; Asghar, M.R.; Russello, G. e-PRNU: Encrypted domain PRNU-based camera attribution for preserving privacy. *IEEE Trans. Dependable Secur. Comput.* **2019**, *18*, 426–437. [CrossRef]
11. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
12. Pang, L.-J.; Wang, Y.-M. A new (t, n) multi-secret sharing scheme based on Shamir's secret sharing. *Appl. Math. Comput.* **2005**, *167*, 840–848. [CrossRef]
13. Jena, R.; Singh, P.; Mohanty, M. SSS-PRNU: Privacy-preserving PRNU based camera attribution using shamir secret sharing. *arXiv* **2021**, arXiv:2106.07029.
14. Mihcak, M.K.; Kozintsev, I.; Ramchandran, K. Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising. In Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258), Phoenix, AZ, USA, 15–19 March 1999; IEEE: Piscataway, NJ, USA, 1999; Volume 6.
15. Mohanty, M.; Zhang, M.; Asghar, M.R.; Russello, G. PANDORA: Preserving privacy in PRNU-based source camera attribution. 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; IEEE: Piscataway, NJ, USA, 2018.