Smartphone Camera De-identification while Preserving Biometric Utility

Sudipta Banerjee and Arun Ross
Michigan State University
{banerj24, rossarun} @cse.msu.edu

Abstract

The principle of Photo Response Non Uniformity (PRNU) is often exploited to deduce the identity of the smartphone device whose camera or sensor was used to acquire a certain image. In this work, we design an algorithm that perturbs a face image acquired using a smartphone camera such that (a) sensor-specific details pertaining to the smartphone camera are suppressed (sensor anonymization); (b) the sensor pattern of a different device is incorporated (sensor spoofing); and (c) biometric matching using the perturbed image is not affected (biometric utility). We employ a simple approach utilizing Discrete Cosine Transform to achieve the aforementioned objectives. Experiments conducted on the MICHE-I and OULU-NPU datasets, which contain periocular and facial data acquired using 12 smartphone cameras, demonstrate the efficacy of the proposed de-identification algorithm on three different PRNU-based sensor identification schemes. This work has application in sensor forensics and personal privacy.

1. Introduction

Sensor identification, or source attribution, [6] refers to the automated deduction of sensor identity from a digital image. Photo Response Non-Uniformity (PRNU) based sensor identification algorithms have been successfully used in the context of RGB [24, 13, 29] and near-infrared (NIR) [18, 20, 2] images. PRNU based sensor identification algorithms typically use image denoising techniques to elicit sensor specific details from an image. The forensic utility of these algorithms has been well established in the literature [14].

Since smartphone devices are intricately linked to their owners, sensor identification using images from smartphone cameras can inevitably lead to person identification. This poses privacy concerns to the general populace [1] and, especially, to photojournalists [28]. Sensor de-identification can mitigate such concerns by removing sensor specific traces from the image. A number of sensor de-identification algorithms, particularly in the context of PRNU suppression, have been developed in the literature [12, 32]. PRNU suppression can be done by either PRNU anonymization or PRNU spoofing.

PRNU anonymization is typically accomplished using strong filtering schemes [11] that perturb the PRNU pattern, or irreversible transformations such as ‘seam carving’ [12] that systematically remove rows and columns from the image. This produces geometrical perturbations which impair correct source attribution. In [23], the authors performed ‘Signature Removal’ by subtracting a scaled version of the PRNU signature (sensor reference pattern) from an image. For an original image, $I$, belonging to sensor $S$, the PRNU anonymized image is obtained as $I' = I - \gamma \hat{K}_S$. Here, $\gamma$ is the scaling factor for the sensor reference pattern $\hat{K}_S$. PRNU spoofing, on the other hand, involves deliberately confounding the image such that a sensor classifier attributes it to a different target sensor rather than the actual source sensor. This can be achieved by injecting the PRNU pattern of the target sensor directly into the test image [15]. The modified image becomes $I' = [I+I \times \gamma \hat{K}_T]$. 

Figure 1: The objective of our work. The original biometric image is modified such that the sensor classifier associates it with a different sensor, while the biometric matcher successfully matches the original image with the modified image.

978-1-7281-1522-1/19/$31.00 ©2019 IEEE
Here, $I$ is the original image and $\hat{K}_T$ is the reference pattern of the target sensor $T$. The term $\gamma$ is a scalar parameter that needs to be empirically tuned to achieve the spoofing. Alternatively, one can first subtract the reference pattern or the ‘fingerprint’ of the source sensor and then insert the target sensor pattern; this is known as ‘Signature Substitution’ [23]. The modified image is represented as $I' = I - \gamma \hat{K}_S + \beta \hat{K}_T$. $I'$ belongs to the source sensor $S$, whose reference pattern is $\hat{K}_S$. The terms $\gamma$ and $\beta$ are scalars which can be optimized through grid-line search.

Majority of the work discussed above focus on images depicting natural scenes. However, for biometric images (face, iris, fingerprints), the perturbations may degrade the quality of the samples making them unsuitable for person recognition. Limited work has been done in the context of sensor de-identification for biometric samples [31]. In [4], the authors proposed an iterative perturbation algorithm, which applied patch-wise modifications to the input biometric image to perform sensor de-identification while retaining its biometric utility. Current methods utilize parameters optimized through exhaustive search techniques and require computation of sensor reference patterns to achieve sensor de-identification [15, 23, 31]. Semi-adversarial learning [27] based methods can be another viable way to accomplish sensor de-identification while retaining biometric matching utility. However, such methods are complex and require a large number of training images. The objective of our work is to develop a rather simple method to perform sensor de-identification, while preserving the biometric recognition utility of the images. The key idea is illustrated in Figure 1. The merits of the proposed method are as follows.

1. Designing a sensor de-identification algorithm that can perform both PRNU anonymization and PRNU spoofing in a non-iterative fashion. This addresses the computational overhead incurred by the algorithms in [11, 4].
2. The proposed de-identification algorithm is applicable to different PRNU estimation schemes and works irrespective of the source and target sensors. This eliminates the need for parameter optimization and computation of the reference patterns corresponding to each pair of source and target sensors as required in [15, 23].
3. The proposed algorithm causes minimal degradation to the biometric content of the images, thus retaining their biometric utility.

The remainder of the paper is organized as follows. Section 2 discusses the concept of PRNU that has been used for sensor identification from images. Section 3 describes the proposed algorithm for PRNU anonymization and PRNU spoofing. In section 4, we describe the dataset, the experimental protocol and the results. Section 5 concludes the paper.

### 2. Photo Response Non-Uniformity (PRNU)

Photo Response Non-Uniformity (PRNU) [24] manifests due to anomalies present in the silicon wafer used during sensor fabrication. PRNU is a type of sensor pattern noise which arises due to the variation of the pixels across the sensor plane to the same light intensity and is intrinsically linked to each sensor. Therefore, PRNU can be used for sensor identification. Sensor identification using PRNU requires the computation of the sensor reference pattern from a set of training images acquired using a specific sensor. When an unknown test image is provided, the noise residual of that image is computed and correlated with the reference pattern of several different sensors. The test image is attributed to the sensor whose reference pattern yields the highest correlation value.

While a number of algorithms have been proposed in the literature to perform sensor identification [3, 25], in this work we will focus on 3 of them — (i) Enhanced PRNU Estimation, (ii) Maximum Likelihood based PRNU Estimation and (iii) Phase PRNU Estimation methods — as these 3 fundamental PRNU based methods have been widely used in the literature. Each method generates the sensor reference pattern from a set of training images as described below.

#### 1. Enhanced PRNU Estimation

The reference pattern is computed as an average of the noise residuals corresponding to the training images [22].

#### 2. Maximum Likelihood based PRNU Estimation (MLE PRNU)

The noise residuals from the training set are weighted by the images and normalized by the sum of the squares of the pixel intensity values of the training images [9], and furthermore, subjected to Wiener filtering and zero-mean operations to remove interpolation artifacts.

#### 3. Phase PRNU Estimation

The phase component of the PRNU is computed in the Fourier domain and then averaged across the training images [19] to generate the reference pattern.

The test noise residual can be extracted by either simply applying a wavelet based denoising filter to the test image or by applying an enhancement model to further attenuate scene influences, as done in Enhanced PRNU Estimation [22].

### 3. Proposed Method

Discrete Cosine Transform (DCT) has been successfully used for lossy image compression [33] or for improving source camera identification [7]. The coefficients located in the top-left portion capture the low frequency components while the bottom-right coefficients encode the high frequency components. Our goal is to modify the images to perturb the PRNU pattern resulting in sensor de-identification. PRNU is a noise-like component which is
metric utility of the image. We then apply the inverse DCT, present in the image. On the other hand, we retain the low frequency components. By suppressing high frequency components, we mask the sensor pattern existing in the image. The scene details are pivotal for biometric recognition. Thus, we ensure preservation of the biometric utility of the image. We then apply the inverse DCT, and the output is the modified image.

**Algorithm 1: PRNU Anonymization.**

**Input:** An image $I$ of size $h \times w$ and parameter $\eta$

**Output:** PRNU anonymized image $I'$

1. Apply 2-dimensional DCT to $I$

2. Compute $m = \min(h, w)$ and $\alpha = \text{round}(\eta \times m)$

3. Extract the high frequency components as follows:

   $$I_{\text{high}} = \text{High}(I_{\text{dct}}, \alpha),$$

   where, the $\text{High}(\cdot, \cdot)$ operator extracts the lower triangular portion of the DCT coefficients along the anti-diagonal direction, regulated by $\alpha$

4. Extract the low frequency components as follows:

   $$I_{\text{low}} = I_{\text{dct}} - I_{\text{high}}$$

5. Apply inverse DCT to obtain the modified image $I' = \text{DCT}^{-1}(I_{\text{low}})$

6. Return the modified image $I'$

**Algorithm 2: PRNU Spoofing.**

**Input:** An image $I$ of size $h \times w$ belonging to source sensor $S$, a set of $N$ candidate images belonging to the target sensor $T$, where each image of size $p \times q$ is denoted as $G^i = [1, \cdots, N]$ and $\eta$

**Output:** PRNU spoofed image $I'$

1. Set $i = 1$

2. Apply 2-dimensional DCT to $I$, $I_{\text{dct}} = \text{DCT}(I)$

3. Extract the low frequency and high frequency components, $I_{\text{low}}$ and $I_{\text{high}}$ as described in Algorithm 1 and set $I_{\text{high}} = 0$

4. Compute $\alpha = \text{round}(\eta \times \min(p, q))$

5. repeat

6. - Apply 2-dimensional DCT to $G^i$, $G^i_{\text{dct}} = \text{DCT}(G^i)$

7. - Extract the high frequency components as follows:

   $$G^i_{\text{high}} = \text{High}(G^i_{\text{dct}}, \alpha)$$

8. - Apply inverse DCT to the high frequency content as follows:

   $$G^i_{\text{high}} = \text{DCT}^{-1}(G^i_{\text{high}})$$

9. - Add the images to generate $T_{\text{high}} + G = G'$ and increment $i = i + 1$

10. until $i = N$

11. Divide by the number of images $T_{\text{high}} = \frac{T_{\text{high}}}{N}$

12. Resize $T_{\text{high}}$ to $h \times w$ using bicubic interpolation

13. Apply inverse DCT to obtain the modified image

14. $I' = \text{DCT}^{-1}(T_{\text{high}} + I_{\text{low}})$

15. Return the modified image $I'$

To achieve sensor de-identification we perform both (i) PRNU Anonymization and (ii) PRNU Spoofing.

**PRNU Anonymization:** Given an image $I$, we first subject it to DCT to yield $I_{\text{dct}}$. We intend to suppress the high frequency information without impairing the low frequency details. To achieve this goal, we define a parameter $\alpha$ that serves as a regulator for high frequency suppression. $\alpha$ is computed as the product of the minimum of the height and width of the image $\min(h, w)$, and a user-defined parameter $\eta$, rounded off to the nearest integer. All DCT coefficients present in the interval $[\text{row} = \alpha : h, \text{col} = \alpha : w]$ are set to zero. Thus, $\alpha$ represents the threshold for the suppression of the DCT coefficients, and that threshold is a function of the image dimensions. We discard the high frequency components and then apply inverse DCT which results in the PRNU anonymized image $I'$. The steps are described in Algorithm 1. The process of PRNU anonymization is illustrated using an example image in Figure 2.

**PRNU Spoofing:** We want the sensor classifier to assign an image belonging to source sensor $S$ to a specific target sensor $T$. To accomplish this task, we perform the following steps.

(i) First, we compute the parameter $\alpha$. Next, we transform the original image $I$ from the source sensor to the DCT domain and then extract its low frequency components (as done in Algorithm 1).

(ii) A set of $N$ candidate images, $G^i$, $i = [1, \cdots, N]$, belonging to the target sensor is selected, and each of them is subjected to DCT resulting in $G^i_{\text{dct}}$ (see Section 4.2). Next, we extract the high frequency coefficients from each $G^i_{\text{dct}}$, apply inverse DCT, and then compute their average to yield $T_{\text{high}}$. This averaged output represents the sensor traces of
Table 1: Dataset specifications. The top block corresponds to MICHE-I dataset [26] and the bottom block corresponds to OULU-NPU face dataset [8]. In the MICHE-I dataset, we denote the brand Apple as ‘Device 1’ and the brand Samsung as ‘Device 2’. Two different smartphones belonging to the same brand and model, e.g., Apple iPhone5, are distinguished as ‘UNIT I’ and ‘UNIT II’.

<table>
<thead>
<tr>
<th>Smartphone Brand and Model</th>
<th>Device Identifier</th>
<th>Sensor</th>
<th>Image Size</th>
<th>Number of Images/Number of Subjects (Training Set)</th>
<th>Number of Images/Number of Subjects (Test Set)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple iPhone 5</td>
<td>Device 1</td>
<td>Front (F)</td>
<td>960×1280</td>
<td>55/7</td>
<td>344/41</td>
</tr>
<tr>
<td></td>
<td>UNIT I</td>
<td>Rear (R)</td>
<td>1536×2048</td>
<td>55/7</td>
<td>355/41</td>
</tr>
<tr>
<td>Apple iPhone 5</td>
<td>Device 1</td>
<td>Front (F)</td>
<td>960×1280</td>
<td>55/6</td>
<td>164/20</td>
</tr>
<tr>
<td></td>
<td>UNIT II</td>
<td>Rear (R)</td>
<td>2448×3264</td>
<td>55/6</td>
<td>170/20</td>
</tr>
<tr>
<td>Samsung Galaxy S4</td>
<td>Device 2</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/5</td>
<td>577/69</td>
</tr>
<tr>
<td></td>
<td>UNIT I</td>
<td>Rear (R)</td>
<td>2322×4128</td>
<td>55/5</td>
<td>600/70</td>
</tr>
<tr>
<td>Samsung Galaxy S6 Edge</td>
<td>—</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/6</td>
<td>0/0</td>
</tr>
<tr>
<td>HTC Desire EYE</td>
<td>—</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/6</td>
<td>0/0</td>
</tr>
<tr>
<td>MEIZU X5</td>
<td>—</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/6</td>
<td>0/0</td>
</tr>
<tr>
<td>ASUS Zenfone Selfie</td>
<td>—</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/6</td>
<td>0/0</td>
</tr>
<tr>
<td>Sony XPERIA C5 Ultra Dual</td>
<td>—</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/6</td>
<td>0/0</td>
</tr>
<tr>
<td>Oppo N3</td>
<td>—</td>
<td>Front (F)</td>
<td>1080×1920</td>
<td>55/6</td>
<td>0/0</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td>660/72</td>
<td>2,210/261</td>
</tr>
</tbody>
</table>

(iii) Finally, we insert the averaged high frequency coefficients into $I$ to generate $I'$ which will now be classified as belonging to the target sensor, resulting in PRNU spoofing.

The implementation details for PRNU spoofing are described in Algorithm 2. The process of PRNU spoofing is illustrated using an example image in Figure 3.

4. Experiments and Results

4.1. Dataset

We used the Mobile Iris Challenge Evaluation (MICHE-I) dataset [26] and the OULU-NPU face dataset [8, 21] for performing the experiments in this work. The MICHE-I dataset comprises of over 3,000 eye images from three devices: Apple iPhone 5, Samsung Galaxy S4 and Samsung Galaxy Tab 2 [14]. However, in our work, we employed the periocular images from two smartphones, Apple iPhone 5 and Samsung Galaxy S4, only. The authors in [14] discovered that two separate units of Apple iPhone 5 were used for data collection. We refer to them as Unit I and Unit II respectively. Further, the images in the dataset were acquired using the front and rear camera sensors, separately. Thus, the MICHE-I dataset used in this work consists of data from 6 sensors. The OULU-NPU face dataset comprises of 4,950 face videos recorded using the front cameras of six mobile devices—Samsung Galaxy S6 Edge, HTC Desire EYE, MEIZU X5, ASUS Zenfone Selfie, Sony XPERIA C5 Ultra Dual and OPPO N3. The videos were recorded in three sessions with different illumination and background scenes. We only use the bonafide face videos/images in the OULU-NPU dataset corresponding to 6 sensors. See Figure 4. The specifications of the dataset are described in Table 1.

We split each dataset into a training set and a test set. We followed a subject-disjoint protocol for creating the training and test sets. The images in the training set are used for generating the reference pattern for each sensor, as indicated in the fifth column of Table 1. Our training set consists of 55 images [5] from each camera sensor in the MICHE-I dataset. The OULU-NPU database contains videos, and so we selected 55 frames (20 frames from the first session, 20 frames from the second, and 15 frames from the third) from 6 subjects, for each of the 6 sensors. The test set comprises of images belonging to the MICHE-I dataset only (see the last column in Table 1). Thus, our dataset consists of 2,870 images corresponding to 333 subjects acquired us-
The experiments evaluated three PRNU estimation schemes: Enhanced PRNU, MLE PRNU and Phase PRNU methods. We used normalized cross-correlation for sensor identification. For the PRNU spoofing experiments, the source and target sensors were from the MICHE-I dataset and were either both front or both rear sensors. Thus, there were $2 \times 2 = 12$ PRNU spoofing experiments. Due to the significant difference in resolutions between the front and rear sensors of smartphones, we did not perform front-to-rear or rear-to-front spoofing. We selected $N$, i.e., the number of candidate images belonging to the target sensor (see Algorithm 2), to be the number of test images for that sensor (see the last column in Table 1).

For the biometric matching experiments, we considered a pericocular matcher, as many of the images used in this work are partial face images. We employed the ResNet-101 architecture pre-trained on ImageNet [10] dataset for performing pericocular matching. We utilized the features from layer 170 which were shown to perform the best for pericocular matching in [17]. We applied Contrast Limited Adaptive Histogram Equalization (CLAHE) to the images before feeding them to the convolutional neural network. We used the cosine similarity for computing the match score between the probe and gallery images. We performed three sets of matching experiments, viz., (i) original: both probe and gallery images comprise of unmodified images, (ii) after: both probe and gallery images comprise of modified images and (iii) cross: the gallery images are the original samples while the probe images are the modified images and the genuine scores are computed by utilizing 2 sample images (belonging to the same subject), i.e., the original image and the modified image; the impostor scores are computed by taking pairs of samples belonging to different subjects. Furthermore, we conducted experiments separately for the two acquisition settings in this database: Indoor and Outdoor.

### 4.3. Results

For the sensor de-identification experiments, we used sensor identification accuracy as the evaluation metric for PRNU anonymization and the spoof success rate (SSR) as the evaluation metric for the PRNU spoofing algorithm. For PRNU anonymization, we first compute the sensor identification accuracy of the original images. Before perturbation, the images are assigned to the correct sensor with high accuracies by all three PRNU estimation schemes (see ‘Original’ column in Table 2). Next, when the sensor classifier accepts the modified images as input, the results indicate a significant degradation in the sensor identification accuracy for a majority of the cases (see ‘After’ column in Table 2). The differences in the sensor identification accuracies before and after perturbation are reported in the ‘Change’ column in Table 2. An average difference (change) of 82.77% in the sensor identification accuracies between pre- and post-perturbed images is observed for all the three PRNU estimation schemes evaluated in this work (Enhanced PRNU: 85.75%, MLE PRNU: 83.20% and Phase PRNU: 79.36%). The results indicate successful PRNU anonymization thereby ensuring sensor de-identification.

The second set of results, pertaining to PRNU spoofing, reports the SSR for the perturbed images. SSR computes the proportion of perturbed images that are assigned to the target sensor. The results in Table 3 indicate successful spoofing with respect to all the PRNU estimation schemes considered in this work. An average SSR of 99.48% is observed when evaluated using Enhanced PRNU, 99.64% when evaluated using MLE PRNU, and 99.28% when evaluated using Phase PRNU for all 12 PRNU spoofing experiments. The proposed spoofing experiment fails to confound the Phase PRNU estimation scheme, particularly when the source sensor is the rear sensor of Device 1 UNIT I and the target sensor is the rear sensor of Device 1 UNIT II. Upon analysis, we observed that the original images belonging to Device 1 UNIT II rear sensor resulted in the lowest sensor identification accuracy for all three PRNU estimation schemes (see Table 2). We speculate that the images may contain some artifacts that are interfering with reliable PRNU estimation as well as the spoofing process. Therefore, we performed another experiment where we increased the value of $\eta$ from 0.7 to 0.9 for that particular spoofing experiment and we observed that the SSR increased to 100% for all 3 PRNU estimation schemes. However, visual analysis reveals that the spoofed images resulting from the two different values of $\eta$ have perceptible differences ($\eta = 0.9$ results in a more blurred image than when $\eta = 0.7$ is used). Finally, we studied the performance of our PRNU spoofing algorithm when a smaller number of candidate images, $N$, were considered in this work.

---

1 We employed Enhancement Model III and we set the user defined threshold to 6 [22, 5].
Table 2: Performance of the proposed algorithm for PRNU Anonymization in terms of sensor identification accuracy (%). Results are evaluated using 3 PRNU estimation schemes. ‘Original’ corresponds to sensor identification using images prior to perturbation. ‘After’ corresponds to sensor identification using images after perturbation and ‘Change’ indicates the difference between the ‘Original’ and ‘After’ sensor identification accuracies. A high positive value in the ‘Change’ field indicates successful PRNU Anonymization.

<table>
<thead>
<tr>
<th>Device Identifier</th>
<th>Sensors</th>
<th>Enhanced PRNU</th>
<th>MLE PRNU</th>
<th>Phase PRNU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Original</td>
<td>After</td>
<td>Change</td>
</tr>
<tr>
<td>Device 1 UNIT I</td>
<td>Front</td>
<td>100.00</td>
<td>97.46</td>
<td>2.54</td>
</tr>
<tr>
<td></td>
<td>Rear</td>
<td>100.00</td>
<td>97.32</td>
<td>2.68</td>
</tr>
<tr>
<td>Device 1 UNIT II</td>
<td>Front</td>
<td>100.00</td>
<td>97.32</td>
<td>2.68</td>
</tr>
<tr>
<td></td>
<td>Rear</td>
<td>100.00</td>
<td>97.32</td>
<td>2.68</td>
</tr>
<tr>
<td>Device 2 UNIT I</td>
<td>Front</td>
<td>100.00</td>
<td>97.32</td>
<td>2.68</td>
</tr>
<tr>
<td></td>
<td>Rear</td>
<td>100.00</td>
<td>97.32</td>
<td>2.68</td>
</tr>
</tbody>
</table>

**AVERAGE** | | 99.48  | 99.64 | 0.97   |

Table 3: Performance of the proposed algorithm for PRNU Spoofing in terms of spoof success rate (SSR). Results are evaluated using 3 PRNU estimation schemes. A high SSR value indicates successful spoofing.

<table>
<thead>
<tr>
<th>Source Sensor</th>
<th>Target Sensor</th>
<th>Enhanced PRNU</th>
<th>MLE PRNU</th>
<th>Phase PRNU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device 1 UNIT I FRONT</td>
<td>Device 1 UNIT II FRONT</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Device 1 UNIT II FRONT</td>
<td>Device 1 UNIT II FRONT</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Device 2 UNIT I FRONT</td>
<td>Device 1 UNIT II FRONT</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Device 1 UNIT II REAR</td>
<td>Device 1 UNIT II REAR</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Device 1 UNIT II REAR</td>
<td>Device 2 UNIT II REAR</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Device 2 UNIT I REAR</td>
<td>Device 1 UNIT II REAR</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>AVERAGE</td>
<td></td>
<td>99.48</td>
<td>99.64</td>
<td>99.28</td>
</tr>
</tbody>
</table>

is employed (50%, 10% and 1% of the test set). Surprisingly, even when only 1% of the test set is used as candidate images, i.e., \( N = 4 \), we observed an average SSR of 99.6% across the three PRNU estimation schemes. However, the spoofed images are significantly degraded as they contain some spurious scene details from the candidate images (possibly, the averaging operation in Step 17 of Algorithm 2 suppresses scene details more aggressively for a high value of \( N \)).

Next, we report the results for the periocular biometric recognition experiments. The periocular matching experiments indicate the preservation of the biometric utility of the images in both PRNU anonymized images and PRNU spoofed images. The ROC curves corresponding to ‘Original’ and ‘After’ matching experiments are within 1% of each other. Figure 5 presents the ROC curves for images subjected to PRNU anonymization. Note that Samsung Galaxy S4 results in overall lower periocular matching performance even for the original images. The cross-matching experiments result in perfect match (100%) for a majority of the cases barring the Samsung Galaxy S4 sensor. The suppression of the high frequency components may also result in removal of edges and other details which can impact the matching performance. For the PRNU spoofing experiments, we have presented the ROC matching curves for each smartphone device or unit used in this work in Figures 6, 7 and 8. The matching experiments show that the perturbation scheme used for PRNU spoofing does not degrade the biometric recognition performance.

5. Conclusion

In this work, we design an algorithm that perturbs a face image acquired using a smartphone camera such that (a) sensor-specific details pertaining to the smartphone camera are suppressed (sensor anonymization); (b) the sensor noise pattern of a different device is incorporated (sensor spoofing); and (c) biometric matching using the perturbed image is not affected (biometric utility). We achieve this by applying the Discrete Cosine Transform to images and further modulating the DCT coefficients to either attain PRNU anonymization or PRNU spoofing. In contrast to existing methods which involve computation of sensor reference
Figure 5: ROC curves for matching **PRNU Anonymized** images. Each row corresponds to a different device identifier: (a) Device 1 UNIT I, (b) Device 1 UNIT II and (c) Device 2 UNIT I.

Figure 6: ROC curves for matching **PRNU Spoofed** images. Here, the source sensor is Device 1 UNIT I. In this case, the target sensors are: (a) Device 1 UNIT II (top row) and (b) Device 2 UNIT I (bottom row).
Figure 7: ROC curves for matching **PRNU Spoofed** images. Here, the source sensor is Device 1 UNIT II. In this case, the target sensors are: (a) Device 1 UNIT I (top row) and (b) Device 2 UNIT I (bottom row).

Figure 8: ROC curves for matching **PRNU Spoofed** images. Here, the source sensor is Device 2 UNIT I. In this case, the target sensors are: (a) Device 1 UNIT I (top row) and (b) Device 1 UNIT II (bottom row).

patterns and exhaustive parameter optimization [15, 23, 31], the proposed method is simple and can achieve highly promising results. In our experiments, we considered face (partial and full) images acquired using the front and rear cameras of different smartphones resulting in data from a total of 12 camera sensors. Our proposed method results in successful camera de-identification for images without compromising the biometric matching performance. An average of \( \approx 82.8\% \) reduction in sensor identification is reported in the case of PRNU anonymization, and an average spoof success rate of \( \approx 99.5\% \) is observed for PRNU spoofing across the three PRNU estimation schemes evaluated in this work.

Future work will consider data from a larger number of sensors to evaluate the proposed algorithm. Moreover, the impact of different denoising and PRNU estimation algorithms will be evaluated to study the robustness of the proposed method. Finally, we will explore methods to make sensor identification schemes resilient to de-identification [30].
References


