Check for updates

ACM DIGITAL LIBRARY | Association for Computing Machinery | acm open

Latest updates: https://dl.acm.org/doi/10.1145/3748266

RESEARCH-ARTICLE

Metrics Matter—Source Camera Forensics for Large-Scale Investigations

SAMANTHA KLIER, University of the Bundeswehr Munich, Neubiberg, Bayern, Germany

HARALD BAIER, University of the Bundeswehr Munich, Neubiberg, Bayern, Germany

Open Access Support provided by:

University of the Bundeswehr Munich

PDF Download
3748266.pdf
16 December 2025
Total Citations: 0
Total Downloads: 80

Published: 13 December 2025
Online AM: 14 August 2025
Accepted: 07 July 2025
Revised: 12 May 2025
Received: 12 May 2025

Citation in BibTeX format

Metrics Matter—Source Camera Forensics for Large-Scale Investigations

SAMANTHA KLIER and HARALD BAIER, Faculty of Computer Science, Research Institute CODE, University of the Bundeswehr Munich, Neubiberg, Germany

Source Camera Forensics (SCF) techniques play a crucial role in digital investigations, particularly for attributing the origin of incriminating content. While traditional SCF methods, such as Sensor Pattern Noise (SPN), perform well in verification tasks, they often fall short for large-scale screening scenarios. This article proposes a shift in evaluation focus, from verification-oriented metrics (e.g., False-Positive Rate), which aim to prevent false convictions, to investigation-oriented metrics (e.g., Recall), which prioritize minimizing evidence loss. To this end, we evaluate three approaches: the classic SPN method; CompaRe, an efficient SPN-based variant; and the Media Source Similarity Hash (MSSH), a non-SPN approach that leverages JPEG structural metadata. On a contemporary dataset, MSSH achieves perfect Recall (1.0), albeit with a lower Precision (0.25). In contrast, the classic SPN and the CompaRe approach reach higher Precision values (up to 0.6 and 0.7, respectively), but their Precision drops below 0.1 for Recall values exceeding 0.7, rendering them unsuitable for the given use case in which evidence preservation is critical. Additionally, MSSH offers a speedup of over 500× compared to SPN-based methods and demonstrates its suitability for large-scale investigations.

CCS Concepts: • **Applied computing** → **Investigation techniques**;

Additional Key Words and Phrases: Digital Forensics, Media Forensics, Source Camera Forensics, Source Camera Identification, SPN, MSSH

ACM Reference format:
Samantha Klier and Harald Baier. 2025. Metrics Matter—Source Camera Forensics for Large-Scale Investigations. *Digit. Threat. Res. Pract.* 6, 4, Article 27 (December 2025), 21 pages.
https://doi.org/10.1145/3748266

1 Introduction

Source Camera Forensics (SCF) is an important area of digital forensics and includes several sub-disciplines, such as determining the brand of a camera used to capture an image [25]. However, assigning an image to a physical device has gained by far the most attention [1, 10, 13, 20, 27, 32]. Recently, it was proposed that two primary phases of SCF should be distinguished, in fact: (i) investigation and (ii) examination [20]. For example, when a large **Child Sexual Abuse Material (CSAM)** server has been seized, along with numerous cameras and potential suspects, the investigation initially benefits from rapid presorting of images based on their source. Later on, in an in-depth examination—when key images and suspects have been identified—the objective is to verify whether a key image was indeed captured with the suspect's camera.

Authors' Contact Information: Samantha Klier (corresponding author), Faculty of Computer Science, Research Institute CODE, University of the Bundeswehr Munich, Neubiberg, Germany; e-mail: samantha.klier@unibw.de; Harald Baier, Faculty of Computer Science, Research Institute CODE, University of the Bundeswehr Munich, Neubiberg, Germany; e-mail: harald.baier@unibw.de.



This work is licensed under Creative Commons Attribution International 4.0.

© 2025 Copyright held by the owner/author(s).
ACM 2576-5337/2025/12-ART27
https://doi.org/10.1145/3748266

For the source verification in forensic examinations, a widely adopted approach is based on the **Sensor Pattern Noise (SPN)**, as introduced by Lukas et al. [21] and further refined by Goljan et al. [15]. This method has remained to be the *de-facto* standard in SCF [20] due to its high accuracy. Later, it was adapted for the use on large-scale databases [16, 23, 35], but the implicit shift in the use case that occurred from verification to identification was not fully embraced. In particular, commonly used performance metrics—such as the **False-Positive Rate (FPR)**—do not align with the priorities of the investigative phase, which primarily aims to minimize the loss of potential evidence [20].

Furthermore, recent studies have shown that SPN-based approaches cannot be reliably applied to post-processed images [2–4, 11]. Here, a particular challenge is the immediate image post-processing performed by smartphones, including various capturing modes and imaging enhancement techniques such as super-resolution [37]. While these techniques are already standard in modern smartphones, they are gradually being introduced to traditional digital cameras as well [8]. Hence, in large-scale investigations, where manual curation of images is not feasible, this poses an increasingly significant problem.

For example, in 2023 alone, roughly 36 million NCMEC reports [26] of CSAM uploads were received, flooding digital forensic laboratories worldwide. Although the investigators still lack adequate technical support, the demanded triage [9, 18, 30] has become a fact for CSAM cases due to limited digital forensic resources and despite the ethical considerations of overlooking victims in CSAM cases, as the lesser of two evils, as pointed out by Casey et al. [9]. Consequently, investigators often just search the well-known Exif metadata of CSAM for camera models associated with the suspect [29], a process that is time-consuming, prone to errors, and easily undermined when metadata has been removed.

Therefore, we evaluate three SCF techniques in the context of an investigative use case and address the gap in appropriate performance evaluation metrics.

Contributions.

(1) We justify the selection of the evaluated approaches, namely the classic SPN [15], CompaRe [6], and **Media Source Similarity Hash (MSSH)** (see Section 3.1).
(2) We argue for the used image set, including images captured in different modes, a high overlap of devices from identical models, and the absence of common metadata, making it particularly challenging to map an image to its source (see Section 3.2).
(3) We highlight the critical importance of Recall for the evaluation in an investigative scenario, where missing a relevant match has more severe consequences than false positives (see Section 3.3).
(4) We evaluate the three approaches based on their (i) overall identification performance, (ii) individual device discrimination, and (iii) robustness on non-standard image modes. We find that the MSSH achieves near-perfect Recall across tests and significantly outperforms SPN-based methods, while the classic SPN approach achieves superior Precision (see Section 4).
(5) We analyze the respective runtime performance both theoretically and empirically, demonstrating that MSSH is dominated by image reading time alone, whereas SPN-based methods involve costly filtering and signal processing steps, making them over 500 times slower in practice (see Section 4.4).

Structure of Article. In Section 1, we introduce the context of SCF investigations, before we review related work in Section 2. Subsequently, we explain our methodology in Section 3 and present the evaluation results and a discussion in Section 4. Finally, we conclude our article in Section 5.

2 Related Work

In this section, we introduce the different purposes of SCF and the associated requirements. Then, we present three SCF approaches, namely, the classic SPN approach, CompaRe as an efficient derivative of SPN, and the MSSH which is not based on the SPN, but on structural metadata.

2.1 Use Case Specific Evaluation of SCF Approaches

A widespread mismatch between the SCF use case that is targeted with efficient approaches and the metrics used for their evaluation has been criticized, by Klier and Baier [20]. In their work, they distinguish between an investigation phase, which may include *identification* and *exploration* approaches, and a subsequent examination phase with the aim of *verification*. Consequently, they propose that approaches should be evaluated according to their purpose, hence, differently.

With regards to the investigation stage, they emphasize the importance of Recall (REC) for identification and Completeness [31] for exploration (e.g., carried out by clustering), respectively, complemented by Precision (PREC) or Homogeneity [31]. In their subsequent analysis, they find that only 5% of the available approaches have been evaluated under that premise. Of these approaches, only Valsesia et al. [34] and Bharathiraja et al. [7] fit the scope of this article. Both are based on the principles of SPN and have been evaluated on the outdated **Dresden Image Database (DIDB)** [14]. However, they are not available for our evaluations. Consequently, there is no SCF approach established for an investigative scenario.

2.2 Classic SPN Approach for Reference

The SPN approach, as initially proposed by Lukas et al. [21] and refined due to Goljan et al. [15], is by far the most studied SCF technique [20]. It was specifically developed for *Verification* purposes and evaluated accordingly, with **False Acceptance Rate (FAR)** and **False-Negative Rate (FNR)**. Until today it is the *de-facto* standard, due to the impressive results of a large-scale evaluation conducted by Goljan et al. [15] back in 2009. They reported a FAR of below $2.4 \cdot 10^{-5}$ and a FNR of less than 0.0238 based on over 1 million images of almost 7,000 devices from 150 digital camera models.[1] However, the computations needed 3 months to conclude, while including only images with less than 6 MP to reduce the computational time.

The underlying principle of the SPN approach is that each sensor has a unique noise pattern which it inevitably introduces to every captured image and can be leveraged for forensic purposes [21]. Basically, Lukas et al. [21] and Goljan et al. [15] use a denoising filter for the reversed purpose, as shown in Equation (1). This means that a common denoising filter F, such as the Wiener Filter [15], is applied to the image I which yields a denoised version of I that can be subtracted from the original to obtain the noise residual N:

$$N(I) = I - F(I). \tag{1}$$

This noise residual can be used to directly compare two images [23], but, more often a so-called *camera fingerprint* is calculated by averaging noise residuals of several non-saturated images. Those so-called *flat* images ideally depict a bright blue sky. In any case, the subsequent comparison between two noise residuals or a noise residual and a *camera fingerprint* is based on cross-correlation of which finally the peak-to-correlation energy ratio [15] is calculated as a single value to express the fitting whereas a value of >60 is considered a match.

However, recent studies have raised concerns about the generalized application of SPN-based methods to images captured by modern smartphones [4, 17], primarily due to the extensive post-processing that occurs immediately after capturing. These concerns are well-founded when considering how the SPN approach operates in contrast to contemporary imaging techniques—such as super-resolution, which merges multiple frames into a single high-resolution image [37]. Notably, this technique is no longer exclusive to smartphones, as it is being adopted in DSLR cameras as well [8].

2.3 CompaRe as Efficient SPN Derivative

Despite these new concerns, the computational cost of the SPN approach has always been an issue. Therefore, several adaptions with the goal to improve its efficiency have been proposed [20], e.g., *CompaRe* by Bernacki [6]. In their evaluation they found that *CompaRe* yields the best results among seven other SPN-based approaches,

[1] downloaded form Flickr.com.

such as Valsesia et al. [35], in terms of runtime and storage efficiency, as well as their effectivity based on **Receiver Operating Characteristics (ROC)**.

Similarly, to the classic SPN approach, *CompaRe* [6] uses a denoising filter to obtain the noise residual from I. However, to reduce further computational and storage costs, subsequently a compact representation of the noise residual is generated and saved. Therefore, the noise residuals are divided into sub-matrices of which the traces are calculated and saved. Consequently, the noise residuals get reduced to a compact size of, e.g., 640×480 px which can be used as input or substitute for other noise based approaches, e.g., Goljan et al. [15] or Mandelli et al. [22].

However, *CompaRe* was not evaluated on images of modern smartphones. This is particularly concerning, as the classic SPN approach, which procedure is similar, yet more accurate, struggles with these devices, as discussed in Section 2.2. Moreover, *CompaRe* was evaluated only in terms of ROC and **Area under Curve (AUC)**. In contrast, the efficient approach of Valsesia et al. [35] was evaluated in terms of Recall and Precision, as expected for an investigative approach, and additionally with ROC and AUC. Hence, it remains unclear whether the performance of *CompaRe* for the intended use case is appropriate.

2.4 MSSH as Efficient Non-SPN Derivative

In stark contrast is the concept of the MSSH [19], making it inherently more reflective of software configurations than hardware characteristics. As a result, the authors deliberately refer to the identification target not as a *camera* or *device*, but more broadly as a *source*.

The MSSH yields a multi-part **Similarity Digest (SD)** of which the first part is determined by the JPEG structure itself, as defined by the JPEG markers in the byte sequence of the file, e.g., `FF C4` for the Define Huffman Table marker. In contrast, the second part represents the markers of the `APP1` segment which stores commonly known metadata, such as Exif. Those markers are extracted in their order of appearance, and a set of n-grams is generated. Subsequently, the set elements are concatenated in alphabetical order which is the SD.

Listing 1. The MSSH source SD of a Samsung Galaxy S9, the two parts are separated by a line break.

```
C0DAC4DBD8E1DAD9DBC0E1E4E4E5E5EBEBC4
0100010101010112010F011001100131011201301 3A011B011B01280128010F013101320132876902130 11A
```

Similarly, to the SPN approach, an SD can be used to directly compare two images, or a so-called *source SD* can be generated based on several images from the same source. For example, Listing 1 shows the source SD from a Samsung Galaxy S9 smartphone. While this source SD is based on several images from the same source, unlike the SPN approach, there is no value in using many similar images. Instead, images taken with different capturing modes should be used. However, one image each is sufficient, as the MSSH is based on a binary feature set, hence, a particular sequence of markers is simply existent or not.

Consequently, the comparison between two SDs is performed using set similarity measures. Specifically, the Jaccard index and the Tversky index have been proposed, with each being particularly well-suited for different tasks: the Jaccard index for direct SD-to-SD comparisons, and the Tversky index for comparisons involving a source SD.

However, a potential vulnerability of the MSSH is missing metadata, e.g., due to deletion by the user to preserve their privacy. Therefore, the MSSH has been evaluated on images in their original state and after deleting all metadata with the result, that the approach is functional in either case. In particular, the MSSH was tested on a dataset with $\approx$ 50,000 images from 287 devices of 189 unique models, containing old DSLR cameras, as well as contemporary smartphones. On this dataset, the MSSH yielded an AUC value of 0.99 which deteriorates only slightly to 0.97, after all metadata has been stripped.

Therefore, the MSSH is a promising candidate for the use in an investigation, but the demanded performance evaluation in terms of RECs and PRECs is missing.

Table 1. Statistics of the Dataset Used for the Evaluation

PrnuModernDevices Dataset	
Date of Publication	2021
Number of Devices	22
Number of Unique Models	17
Number of Images	550
Number of JPEGs	520
Types of Images	Flat, natural, bokeh
Number of Images for Reference Generation	154
Number of Images for Evaluation	366

3 Methodology

In this section, we present the methodology of our evaluation and introduce the implementation details of the selected approaches, and argue for the dataset and the performance metrics used.

3.1 Implementations and Execution

We select the classic SPN approach (see Section 2.2), CompaRe (see Section 2.3), and the MSSH (see Section 2.4) due to the following reasons. Firstly, the classic SPN approach is the *de-facto* standard for SCF and serves as a well-studied reference with unprecedented accuracy. Secondly, the CompaRe approach is supposedly the best efficient derivative. Thirdly, the MSSH is a promising, technically alternative approach that is tailored for pre-processing purposes.

For the evaluations, the openly available source codes of the classic SPN approach[2] and the MSSH[3] have been used, while the CompaRe implementation used by Bernacki [6] has been made available to us upon request. For the MSSH two different functions to compare SDs have been proposed, in particular, the Jaccard and the Tversky Index. Generally, the latter of the two is advised to be used if a source SD can be generated, as it is the case in our scope. Hence, we use the Tversky Index of the MSSH throughout our experiments.

The execution environment is a commodity hardware laptop, running Windows 11 Pro on a Core i7 CPU, with 32 GiB RAM and an SSD. Also, no parallelization or other performance optimizations have been introduced to the available source codes and all calculations are performed on the CPU.

3.2 Dataset

For our evaluation we select the PrnuModernDevices dataset [17] which is one of the more recently published datasets. We list its main statistics in Table 1. Overall, the 2021 published dataset contains 550 images, of which 520 are JPEGs, from 22 devices of 17 smartphone models.

Although PrnuModernDevices is a rather small dataset comprising 550 images, it provides key advantages over other datasets for our setting. For example, there is a high overlap of devices from the same models (i.e., 29%), which is important for assessing the ability to differentiate individual devices. Moreover, to analyze the general applicability of an approach on images stemming from a smartphone, images taken in different capturing modes are available. It also provides "flat" images to generate a strong reference pattern for the SPN approaches.

Therefore, the FloreView [5] dataset, which is newer (i.e., published 2023) and bigger (6,637 images), is inferior in both other aspects, with a device-model overlap of only 9% and providing images captured only in standard mode. In contrast, while the large (11,732 images) and widely used VISION [33] dataset, has a significant device-model

[2]https://dde.binghamton.edu/download/camera_fingerprint/.
[3]https://github.com/SamKlier/mssh.

overlap of 20%, it is from 2017 and contains only images of smartphones before the era of AI-based imaging. Consequently, while the PrnuModernDevices dataset is small, it enables the evaluation of the most critical aspects of SCF better than other larger datasets.

However, due to the constraint that the MSSH is only applicable on JPEG files, 30 non-JPEG files had to be excluded. Furthermore, the MSSH may be vulnerable to the deletion of common metadata, such as Exif, for which tools are widespread and easy to use. Therefore, in the upcoming evaluations, the MSSH is always applied to the images in their original state and without any metadata,[4] as denoted by the *without metadata* affix. However, such anti-forensic measures were not introduced for the SPN based approaches, as tools that, e.g., introduce misleading noise, are not widespread.

Reference Generation. Consequently, the references of a camera or source are generated, as imposed by each approach. In particular, this means that for the classic SPN and CompaRe all "flat" images per device have been used to generate a noise based reference pattern by averaging. In contrast, for the reference pattern of the MSSH, the file container structures of one "flat," one "nat" and one "bokeh" image per device were used and combined in one feature set for the generation of the source SD. Each image that was used for any reference generation has been excluded from the subsequent evaluations.

3.3 Evaluation Metrics

As we target at investigative approaches we make use of both Recall (see Equation (3), also referred to as **True-Positive Rate (TPR)**) and Precision (see Equation (2)) as evaluation metrics as proposed by Klier and Baier [20]. In our context, the aim of a high Recall value is to minimize evidence loss. In contrast, a suitable Precision ensures that only a manageable amount of unrelated images (false-positives) need to be investigated further. In contrast, the FPR is a key metric for *Verification* due to the utmost importance of preventing false convictions. However, at this stage, no convictions are made and large data volumes are only reduced, hence, the FPR is dispensable. However, they found that only Recall/TPR is commonly used, mostly alongside the FPR which are the two metrics considered by ROC curves of which often the AUC is used as a single metric, to evaluate the performance [20].

The general work on ROCs in contrast to **Precision Recall Curves (PRCs)** of Davis and Goadrich [12] also states that ROCs can be misleading and PRCs may be more informative, depending on the use case. Consequently, they advocate for a "context aware" selection of evaluation metrics which is along the lines of Klier and Baier [20]. McDermott et al. [24] generally agree with their recent work on AUC and **Area under Precision Recall Curve (AU-PRC)**, but argue that the AU-PRC, as a single metric to determine model performance, should not be generally used, as it favors inadvertently higher-prevalence sub-groups. Therefore, they conclude that Precision and Recall can and shall be considered, depending on the use case, yet, the AU-PRC should not replace the AUC as a single performance metric.

While these works have emphasized the limitations of ROC and AUC in investigative contexts—particularly due to their poor reflection of operational relevance—we include them here for the sake of comparability with prior literature. However, in alignment with the goals of investigation-driven SCF, we place greater interpretive weight on Recall and Precision, which more directly capture the need for evidence completeness and triage feasibility. Accordingly, PRC-plots are emphasized in our analysis and discussion. However, the results are not evaluated in terms of AU-PRC to avoid favoring the higher prevalent sub-groups. We use the following evaluation metrics throughout this study:

—*Precision*: The proportion of true positives among all positive predictions:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{2}$$

[4]Metadata was stripped with ExifTool's "-all=" command.

—*Recall/TPR*: The proportion of true positives identified out of all actual positives:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{3}$$

—*FPR*: The proportion of false positives among all actual negatives:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}. \tag{4}$$

—*ROC curve*: A plot of the TPR against the FPR at various threshold settings.
—*AUC*: The area under the ROC curve, providing a single measure of performance across all classification thresholds.
—*PRC*: A plot of Precision against Recall for different threshold values. Additionally, we provide the baseline for the PRCs which indicates the results of a random guessing approach and is the positive ration of the used dataset.

The concrete calculations of the evaluation metrics were executed with scikit-learn[5] which provides a function for each given metric and curve and automatically determines the tested thresholds.

4 Evaluation

We begin by evaluating the performance of all approaches on the complete dataset. Next, we examine whether the MSSH's ability to distinguish individual devices is indeed inferior to that of SPN-based methods, as suggested by its design. We then analyze the impact of different capture modes on the performance of each method and assess their runtime efficiency both theoretically and empirically. Finally, we evaluate the worst-case scenario for MSSH to stress-test the most promising candidate for use in investigative contexts.

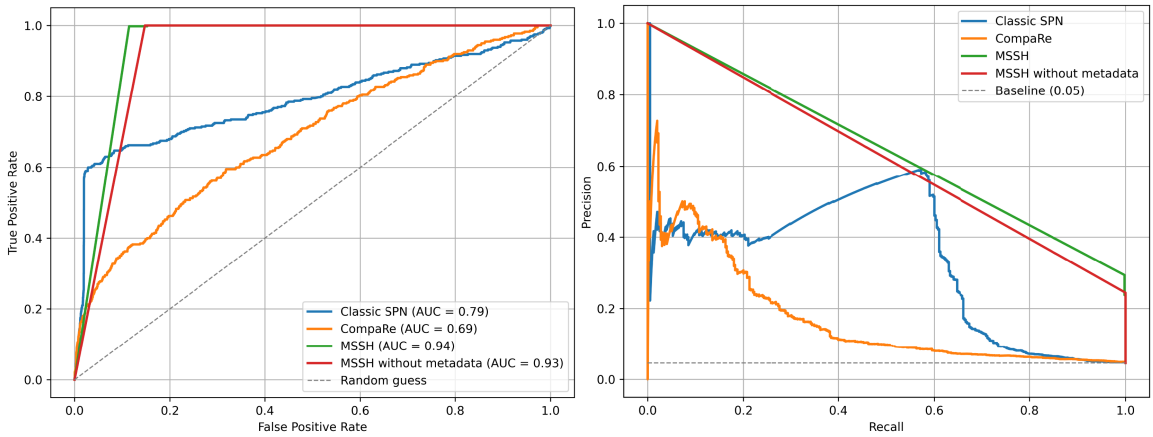
4.1 Overall Identification Performance

First, the overall performance on the complete dataset is evaluated, as shown in Figure 1. On the left side (Figure 1(a)), the ROC curve is shown and the corresponding AUC value is plotted in its legend. The more important, use case specific performance, in terms of PRC is shown on the right side (Figure 1(b)). As shown, all methods clearly surpass the baseline levels.

Curves of the MSSH. Notably, the curves of the MSSH appear unusual due to the dominance of straight lines. For example in Figure 1(a), the red curve of the MSSH jumps straight to a TPR of 1.0; however, the green curve exhibits a small notch, before reaching a TPR of 1.0. The same can be observed for the PRCs, e.g., in Figure 1(b). Please refer to Appendix Tables A1 and A2, for a detailed excerpt of the MSSH results. However, these results can be explained, as per definition, the ROCs and PRCs start and end at $(0.0, 0.0)/(1.0, 1.0)$ and $(0.0, 1.0)/(1.0, \texttt{baseline})$, respectively. Also, the MSSH yields no values in the low TPR, respectively Recall regions, hence the jump or straight decline. Finally, once reached, the high TPR/Recall values are retained. Again, this results in a mostly straight line. Therefore, while the curves appear unusual, they can be explained by the fact, that the MSSH shows little dependency on the used thresholds. However, the curves have to be interpreted with caution, as, e.g., a Precision of > 0.3 in Figure 1(b) was not achieved by the MSSH.

PRC. Overall, the MSSH demonstrates a robust Recall of ≈ 1.0 and a Precision of ≈ 0.25, whereas varying the threshold has minimal effect on the outcome of the MSSH. Notably, even in the absence of metadata, the method remains its performance mostly, as shown by the performance of the *MSSH without metadata* curve. In contrast, the classic SPN approach exhibits a distinct performance sweet spot, with both Recall and Precision peaking around ≈ 0.6. Before this point, both metrics show a weaker performance. Beyond this point, Precision declines so sharply that any further gain in Recall comes at an impractically high cost, rendering the tradeoff

[5] https://scikit-learn.org/, version: 1.5.1.



(a) ROC, for all devices.

(b) PRC, for all devices.

Fig. 1. Evaluation results of the complete PrnuModernDevices dataset.

practically ineffective. Finally, CompaRe shows relatively strong Precision at very low Recall values, comparable to the classic SPN and exceeding the MSSH, but its Precision falls below that of the MSSH for Recall values above ≈ 0.2.

Therefore, CompaRe performs the weakest among the evaluated approaches, particularly with respect to the most critical metric, Recall. In contrast, the MSSH is the only approach that achieves a perfect Recall of 1.0 with a PREC of >0.25 which is significantly above the baseline, while the other approaches show hardly a benefit to guessing for Recall values above 0.8. However, while the MSSH is robust and easy to deploy, as it eliminates the need for a delicate calibration process, it also implies a lack of flexibility, as the absence of a tradeoff between Recall and Precision prevents adaptation to specific investigative scenarios. On the contrary, the classic SPN shows receptivity for different thresholds which suggest that it is tunable in theory. However, it has a narrow optimal operating range in practice, which indicates a lack of robustness.

ROC Curve. Regarding the AUC (see Figure 1(a)), the MSSH performs best with distinction and its effectiveness remains nearly unchanged in the absence of metadata, as indicated by the comparable AUC values of 0.94 versus 0.93. In contrast, both the classic SPN approach and CompaRe exhibit substantially lower AUC values, respectively, 0.79 and 0.69. However, both AUC values are far below the values reported for these approaches on older datasets, e.g., 0.99 [33] for the classic SPN approach and 0.93 for CompaRe, respectively, on the VISION dataset or a subset of the DIDB, but match with the observations of Iuliani et al. [17] on this dataset. However, the ROC shows that meaningful low FPRs can only be achieved with the classic SPN which matches the use case of *Verification* for what it is originally intended.

4.2 Individual Source Identification

Unlike the SPN-based approaches, the MSSH cannot distinguish hardware devices by design, but may be able to differentiate distinct source cameras by their configured software. Therefore, it remains unclear to which extent certain cameras, e.g., of the same model, can be distinguished by the software traces used. To analyze the ability of the MSSH to differentiate devices and models, the source SDs of each device are evaluated in terms of uniqueness. Therefore, in Table 2, each compartment symbolizes one distinct source SD. For instance, the two devices of the Huawei PSmart2019 have a unique SD whereas the P10, the three P20pro and the P30lite all share the same SD. Notable devices that share the same source SD have been identified to be challenging for the SPN approach [17], as well.

Table 2. List of Devices in the PRNU Modern Devices Dataset, Compartmentalized per Unique MSSH SD, Sorted by Brand

Brand	Model	Device
Apple	iPhone11	C20
	iPhone11	C21
Apple	iPhone11ProMax	C22
Apple	iPhoneX	C19
Huawei	P30lite	C01
	P20pro	C02
	P20pro	C03
	P20pro	C04
	P10	C09
Huawei	PSmart2019	C05
Huawei	PSmart2019	C06
Huawei	P20lite	C07
	P20lite	C08
Samsung	GalaxyS6	C13
Samsung	GalaxyS9	C14
Samsung	GalaxyS9+	C15
Samsung	GalaxyA70	C16
OnePlus	6T	C17
	6	C18
Xiaomi	MiNote10	C10
Xiaomi	RedmiNote8T	C11
	MiA3	C12

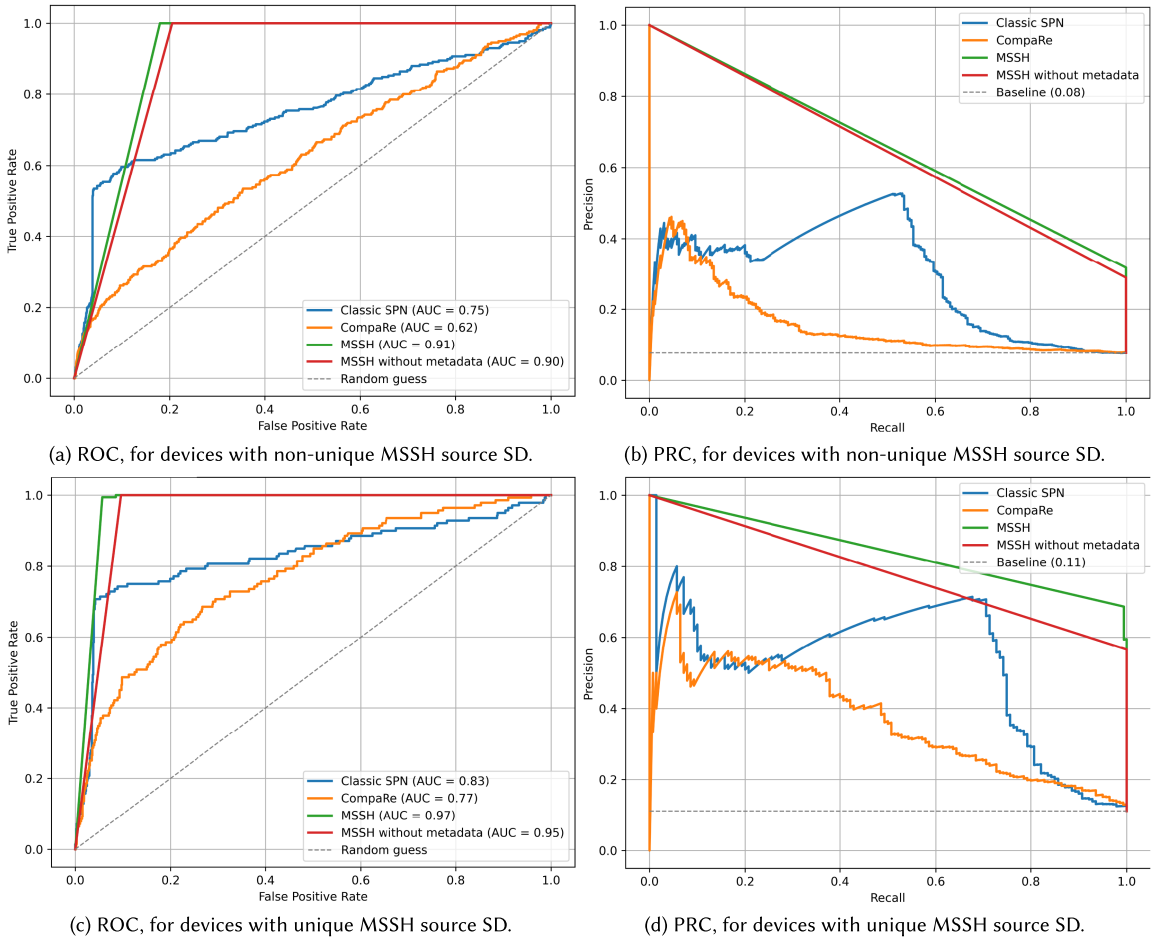
Therefore, the approaches have been evaluated on two subsets (see Figure 2), containing (i) all devices with a non-unique source SD (see Figure 2(a) and (b)), and (ii) all devices with a unique SD (see Figure 2(c) and (d)).

Overall, the results are along the lines of Section 4.1. However, the subset of devices with non-unique source SDs is per design particularly challenging for the MSSH, but the performance of the other approaches deteriorates even more. In contrast, on the subset with devices that have a unique source SD the performance of the MSSH improves significantly in terms of Precision with retained near perfect Recall whereas the performance of the classic SPN an CompaRe improves evenly across Recall and Precision. Consequently, the devices that are problematic for the MSSH in this dataset are also problematic in terms of their SPN and vice-versa.

4.3 Identification on Bokeh and Natural Images

Due to the fact that bokeh images are known to be problematic for the SPN approach, we evaluate the approaches on two further subsets (see Figure 3), containing all images taken in standard capturing mode (see Figure 3(a) and (b)), and all images taken in bokeh mode (see Figure 3(c) and (d)).

Here, the results of the MSSH are almost unaffected by the different capturing modes and also in comparison to the results of Section 4.1. However, the performance differences of the classic SPN approach are tremendous and collapses for bokeh images, in contrast to images taken in the standard capturing mode. Although for these images a Recall of >0.8 is achievable with meaningful PREC values of 0.6, the decline in Precision is rapid again and close to baseline for very high Recall values exceeding 0.9. Interestingly, CompaRe is hardly affected and retains



(a) ROC, for devices with non-unique MSSH source SD.

(b) PRC, for devices with non-unique MSSH source SD.

(c) ROC, for devices with unique MSSH source SD.

(d) PRC, for devices with unique MSSH source SD.

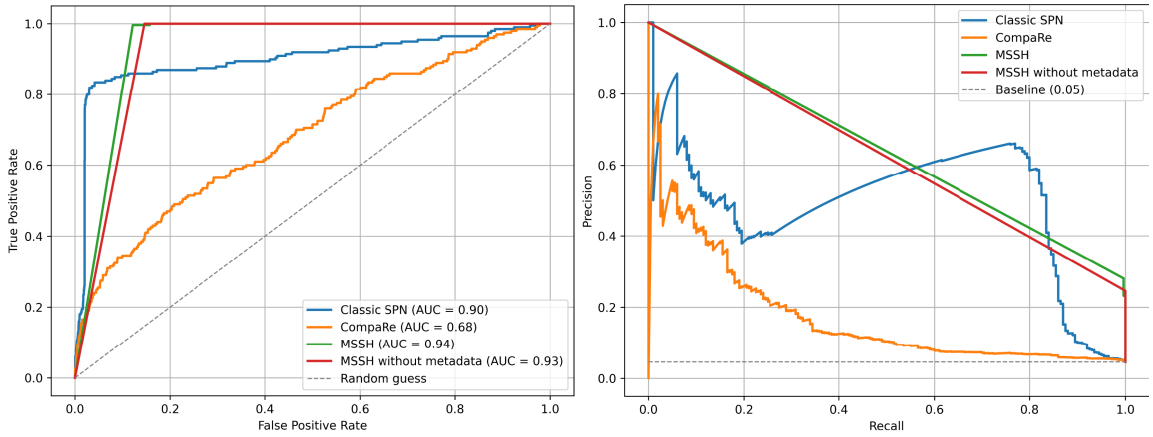
Fig. 2. Contemplation of subsets to evaluate the performance on problematic devices.

its performance for bokeh images which, however, is already at a low level. Therefore, while the performance of the MSSH is agnostic to the tested capturing modes, the classic SPN approach provides a clear advantage in terms of Precision for images taken in standard capturing mode.

4.4 Evaluation of Runtime Efficiency

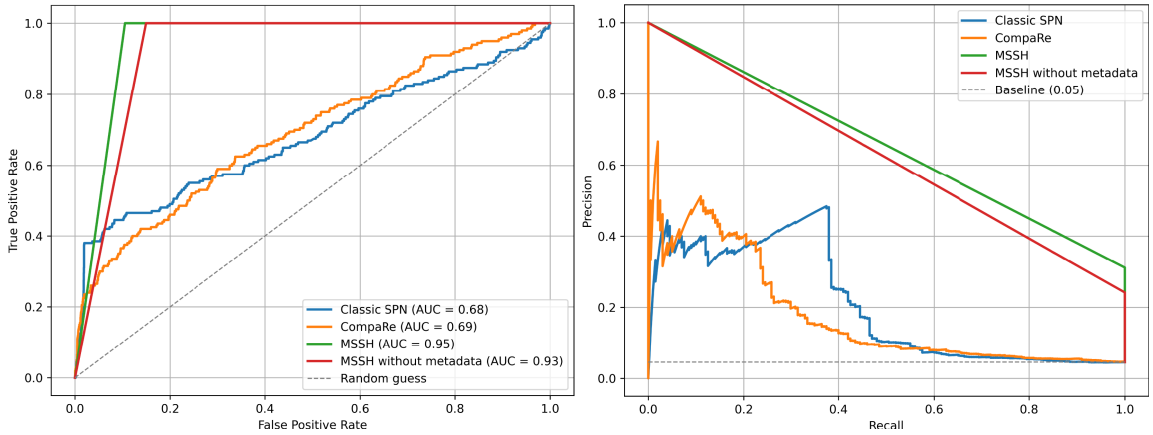
The runtime efficiency of the approaches varies notably due to their distinct computational dependencies. In any case, an image with a file size of b Bytes has to be read from storage.

Subsequently, the MSSH extracts the used features directly and linearly from the byte stream, yielding a runtime complexity of $O(b)$. In contrast, the classic SPN and CompaRe operate on the pixels, to extract the noise and, hence, require decoding with $O(N)$ [36] (N being the resolution) of the visual content first. Subsequently, in the extraction phase, their performance is primarily governed by the noise filtering operations, which are computationally intensive and also depend on image resolution. For example, an efficient implementation of the Wiener Filter has a runtime complexity of $O(N\ log(N))$ [28]. During the comparison phase, the classic SPN relies on 2D cross-correlation over the original image dimensions, resulting in a complexity of $O(N\ log(N))$,



(a) ROC, for images taken with the standard capturing mode.

(b) PRC, for images taken with the standard capturing mode.



(c) ROC, for bokeh images.

(d) PRC, for bokeh images.

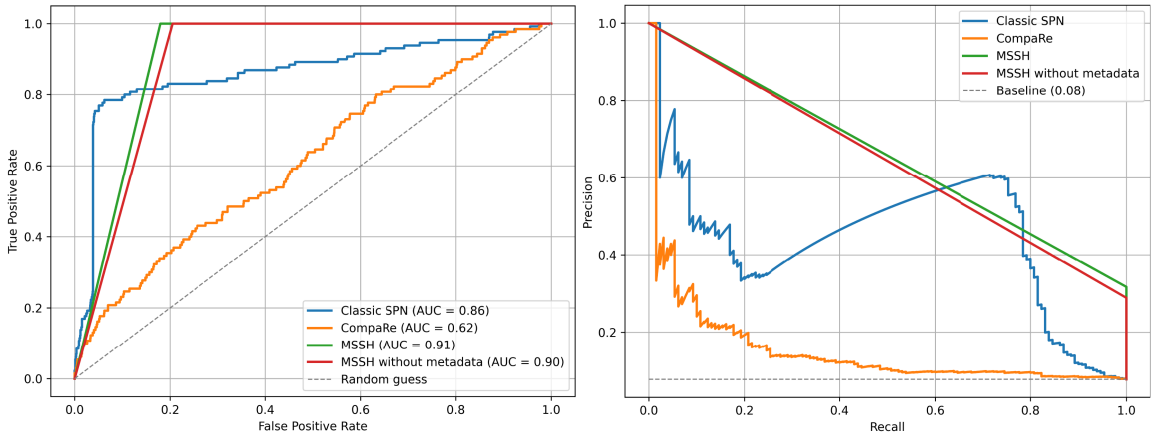
Fig. 3. Contemplation of subsets to evaluate the performance on images taken with different capturing modes.

whereas CompaRe also performs a cross-correlation, but on fixed-sized downsampled images e.g., (640×480 px), leading to a constant complexity of $O(1)$. In contrast, the MSSH compares short digests with set operations, which results also in a complexity of $O(1)$.

In practice, this means that all approaches read the image from storage, but, only for the MSSH this phase is dominating the runtime efficiency. The other approaches further need several steps of signal processing operations. Therefore, in our experiments on a commodity laptop, the MSSH generated its reference patterns for all 22 devices in $\approx$ 15 seconds whereas the classic SPN and CompaRe needed $\approx$ 23 minutes, respectively. The subsequent comparisons of each image to the reference patterns were concluded in 63 hours, 51 hours, and 6 minutes, respectively, by the classic SPN, CompaRe, and MSSH. Therefore, MSSH is over 500 times faster than CompaRe and over 600 times faster than the classical SPN approach.

4.5 MSSH Worst Case

Our findings underscore the limitations of classic SPN and its efficient derivative, CompaRe, in investigative preprocessing scenarios. Specifically, these approaches fail to achieve a REC of approximately 1.0 without sacrificing



(a) ROC, worst-case MSSH.

(b) PRC, worst-case MSSH.

Fig. 4. Evaluation of the worst case for the MSSH, and hence, only images taken in standard mode of devices with non-unique MSSH SD are included.

PREC to unusable levels in most scenarios. This shortcoming, combined with prohibitive runtimes—often in the order of weeks or months for realistically sized datasets—renders them impractical for large-scale investigations where timely data reduction without evidence loss is critical. In contrast, the MSSH consistently demonstrates a REC of 1.0 across all evaluated scenarios, including challenging subsets, and does so with runtimes bounded by the speed the image is read from storage.

Therefore, we evaluate the robustness of the MSSH's REC even further by considering the worst case: A subset composed solely of images taken in the standard capturing mode from devices with non-unique source SDs, as shown in Figure 4. However, even under this condition, the MSSH maintains a perfect REC of 1.0. But, this perfect REC throughout our evaluations comes at the cost of a consistently low PREC. This means that a substantial number of false positives are inevitably introduced alongside the true matches which can be a problem. However, if the MSSH proves to yield a robust REC of $\approx$ 1.0 on other and larger studies, it will be a valuable addition for initial filtering stages in large-scale investigations, where completeness of evidence must be preserved and processing must scale.

5 Conclusion and Future Work

The domain of SCF is dominated by approaches used to verify that an image stems from a specific camera with high certainty. However, the differing demands in the investigate phase are overlooked. Thus, we identified two promising candidates for this purpose, namely CompaRe and MSSH, and compared their performance to the *de-facto* standard for SCF, the classic SPN approach. Most notably, the MSSH is the only approach that achieves a Recall of 1.0 which is the most critical metric for a pre-processing step in an investigation. Moreover, our evaluation shows that the MSSH outperforms the classic SPN approach also in terms of Precision for newer smartphones, in most scenarios. This is surprising, as the MSSH is the only approach that does not distinguish between hardware devices by design, but software configurations. However, even when the worst case for the MSSH is considered, CompaRe in comparison is ineffective, while the classic SPN approach is only superior in low FPR areas which are not the main concern in an investigation. Additionally, it is evident that the runtime performance of the SPN approaches is significantly inferior to the MSSH.

However, enhancing the Precision of the MSSH would be beneficial under the constraint that a Recall/TPR of $\approx$ 1.0 can be sustained. Moreover, the combination with more accurate approaches, such as the classic SPN

approach, should be studied. Unfortunately, the two evaluated SPN-based approaches encounter issues with the same cameras as the MSSH, thereby indicating that a combination of both approaches, as is, may not be a satisfactory solution. Therefore, SPN-based approaches must be enhanced or superseded for contemporary devices.

Yet, while this work suggests that the MSSH is a robust and practical pre-processing approach for real-world scenarios, experiments with larger and ideally more realistic datasets are needed.

References

[1] Younes Akbari, Somaya Al-Maadeed, Omar Elharrouss, Fouad Khelifi, Ashref Lawgaly, and Ahmed Bouridane. 2022. Digital forensic analysis for source video identification: A survey. *Forensic Science International: Digital Investigation* 41 (2022), 301390.

[2] Omar Al Shaya, Pengpeng Yang, Rongrong Ni, Yao Zhao, and Alessandro Piva. 2018. A new dataset for source identification of high dynamic range images. *Sensors* 18, 11 (2018), 3801.

[3] Chiara Albisani, Massimo Iuliani, and Alessandro Piva. 2021. Checking PRNU usability on modern devices. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '21)*. IEEE, 2535–2539.

[4] Daniele Baracchi, Massimo Iuliani, Andrea G. Nencini, and Alessandro Piva. 2021. Facing image source attribution on iPhone X. In *19th International Workshop on Digital Forensics and Watermarking (IWDW '20)*. Springer, 196–207.

[5] Daniele Baracchi, Dasara Shullani, Massimo Iuliani, and Alessandro Piva. 2023. FloreView: An image and video dataset for forensic analysis. *IEEE Access* 11 (2023), 109267–109282. DOI: https://doi.org/10.1109/ACCESS.2023.3321991

[6] Jarosław Bernacki. 2022. Digital camera identification by fingerprint's compact representation. *Multimedia Tools and Applications* 81, 15 (2022), 21641–21674.

[7] S. Bharathiraja, B. Rajesh Kanna, and M. Hariharan. 2023. A deep learning framework for image authentication: An automatic source camera identification Deep-Net. *Arabian Journal for Science and Engineering* 48, 2 (2023), 1207–1219.

[8] Canon. 2023. Deep Learning Image Processing Technology. Retrieved November 30, 2023 from https://global.canon/en/technology/dl-iptechnology-2023.html

[9] Eoghan Casey, Monique Ferraro, and Lam Nguyen. 2009. Investigation delayed is justice denied: Proposals for expediting forensic examinations of digital evidence. *Journal of Forensic Sciences* 54, 6 (2009), 1353–1364.

[10] Ivan Castillo Camacho and Kai Wang. 2021. A comprehensive review of deep-learning-based methods for image forensics. *Journal of Imaging* 7, 4 (2021), 69.

[11] Mo Chen, Jessica Fridrich, Miroslav Goljan, and Jan Lukás. 2008. Determining image origin and integrity using sensor noise. *IEEE Transactions on Information Forensics and Security* 3, 1 (2008), 74–90.

[12] Jesse Davis and Mark Goadrich. 2006. The relationship between precision-recall and ROC curves. In *23rd International Conference on Machine Learning*, 233–240.

[13] Zeno J. Geradts, Jurrien Bijhold, Martijn Kieft, Kenji Kurosawa, Kenro Kuroki, and Naoki Saitoh. 2001. Methods for identification of images acquired with digital cameras. In *Enabling Technologies for Law Enforcement and Security*, Vol. 4232. SPIE, 505–512.

[14] Thomas Gloe and Rainer Böhme. 2010. The "Dresden image database" for benchmarking digital image forensics. In *2010 ACM Symposium on Applied Computing*, 1584–1590.

[15] Miroslav Goljan, Jessica Fridrich, and Tomáš Filler. 2009. Large scale test of sensor fingerprint camera identification. In *Media Forensics and Security*, Vol. 7254. SPIE, 170–181.

[16] Miroslav Goljan, Jessica Fridrich, and Tomáš Filler. 2010. Managing a large database of camera fingerprints. In *Media Forensics and Security II*, Vol. 7541. SPIE, 75–86.

[17] Massimo Iuliani, Marco Fontani, and Alessandro Piva. 2021. A leak in PRNU based source identification—Questioning fingerprint uniqueness. *IEEE Access* 9 (2021), 52455–52463.

[18] Da-Yu Kao, Ni-Chen Wu, and Fuching Tsai. 2020. A triage triangle strategy for law enforcement to reduce digital forensic backlogs. In *2020 22nd International Conference on Advanced Communication Technology (ICACT)*. IEEE, 1173–1179.

[19] Samantha Klier and Harald Baier. 2025. Media source similarity hashing (MSSH): A practical tool for large-scale media investigations. *Forensic Science International: Digital Investigation* (2025).

[20] Samantha Klier and Harald Baier. 2025. Source camera identification—Do we have a gold standard? *Forensic Science International: Digital Investigation* 52 (2025), 301858.

[21] J. Luka, J. Fridrich, and M. Goljan. 2006. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security* 1, 2 (2006), 205–214.

[22] Sara Mandelli, Davide Cozzolino, Paolo Bestagini, Luisa Verdoliva, and Stefano Tubaro. 2020. CNN-based fast source device identification. *IEEE Signal Processing Letters* 27 (2020), 1285–1289.

[23] Francesco Marra, Giovanni Poggi, Carlo Sansone, and Luisa Verdoliva. 2017. Blind PRNU-based image clustering for source identification. *IEEE Transactions on Information Forensics and Security* 12, 9 (2017), 2197–2211.

[24] Matthew McDermott, Haoran Zhang, Lasse Hansen, Giovanni Angelotti, and Jack Gallifant. 2024. A closer look at AUROC and AUPRC under class imbalance. In *Advances in Neural Information Processing Systems*, Vol. 37, 44102–44163.
[25] Patrick Mullan, Christian Riess, and Felix Freiling. 2020. Towards open-set forensic source grouping on JPEG header information. *Forensic Science International: Digital Investigation* 32 (2020), 300916.
[26] National Center for Missing & Exploited Children. 2024. 2023 CyberTipline Reports by Electronic Service Providers (ESP). Retrieved from https://www.missingkids.org/content/dam/missingkids/pdfs/2023-reports-by-esp.pdf
[27] Chijioke Emeka Nwokeji, Akbar Sheikh-Akbari, Anatoliy Gorbenko, and Iosif Mporas. 2024. Source camera identification techniques: A survey. *Journal of Imaging* 10, 2 (2024), 31.
[28] Alan V. Oppenheim and Ronald W. Schafer. 2009. *Discrete-Time Signal Processing* (3rd. ed.). Prentice Hall, Upper Saddle River, NJ.
[29] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. Garcıa Villalba, and J. C. Hernandez-Castro. 2013. Techniques for source camera identification. In *6th International Conference on Information Technology*, 1–9.
[30] Marcus K. Rogers, James Goldman, Rick Mislan, Timothy Wedge, and Steve Debrota. 2006. Computer forensics field triage process model. *Journal of Digital Forensics, Security and Law* 1, 2 (2006), 2.
[31] Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 410–420.
[32] Husrev Taha Sencar, Luisa Verdoliva, and Nasir Memon. 2022. *Multimedia Forensics*. Springer Nature.
[33] Dasara Shullani, Marco Fontani, Massimo Iuliani, Omar Al Shaya, and Alessandro Piva. 2017. Vision: A video and image dataset for source identification. *EURASIP Journal on Information Security* 2017, 1 (2017), 1–16.
[34] Diego Valsesia, Giulio Coluccia, Tiziano Bianchi, and Enrico Magli. 2015. Compressed fingerprint matching and camera identification via random projections. *IEEE Transactions on Information Forensics and Security* 10, 7 (2015), 1472–1485.
[35] Diego Valsesia, Giulio Coluccia, Tiziano Bianchi, and Enrico Magli. 2015. Large-scale image retrieval based on compressed camera identification. *IEEE Transactions on Multimedia* 17, 9 (2015), 1439–1449.
[36] Gregory K. Wallace. 1991. *The JPEG Still Picture Compression Standard*, Vol. 34. ACM, 30–44 pages.
[37] Bartlomiej Wronski, Ignacio Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai Liang, Marc Levoy, and Peyman Milanfar. 2019. Handheld multi-frame super-resolution. *ACM Transactions on Graphics* 38, 4 (2019), 1–18.

Appendix

A Exemplary Results of MSSH

A.1 MSSH ROC Values

Table A1. TPR and FPR Yielded by the MSSH, for the Given Threshold, on Images with Metadata, as Presented in Figure 1(a)

TPR	FPR	Threshold
0.9978165938864629	0.11436889166146808	1.0
0.9978165938864629	0.11717612809315867	0.9688
0.9978165938864629	0.12778124350176753	0.96
0.9978165938864629	0.14743189852360158	0.9583
0.9978165938864629	0.1486795591599085	0.95
0.9978165938864629	0.1496153046371387	0.9444
0.9978165938864629	0.15242254106882927	0.9394
1.0	0.15242254106882927	0.9375
1.0	0.15335828654605946	0.9286
1.0	0.16895404449989604	0.9231
1.0	0.18954044499896028	0.92
1.0	0.19089207735495944	0.9048
1.0	0.1909960490746517	0.9
1.0	0.1926595965897276	0.8966
1.0	0.19359534206695778	0.8929
1.0	0.1981700977334165	0.8824
1.0	0.19993761696818466	0.88
1.0	0.20014556040756915	0.8667
1.0	0.20180910792264503	0.8621
1.0	0.20212102308172178	0.85
1.0	0.2048242877937201	0.8421
1.0	0.2066957787481805	0.8333
1.0	0.21085464753587024	0.8095
1.0	0.21303805364940737	0.8
1.0	0.2205240174672489	0.7778
1.0	0.22634643377001457	0.7619
1.0	0.2295695570804741	0.75
1.0	0.24038261592846746	0.7368
1.0	0.24225410688292784	0.7222
1.0	0.2430858806404658	0.7143
1.0	0.25493865668538157	0.7
1.0	0.2601372426699938	0.6923
1.0	0.260241214389686	0.6875
1.0	0.27375753794967767	0.6842
1.0	0.2893532959035142	0.68

(Continued)

Table A1. Continued

TPR	FPR	Threshold
1.0	0.29777500519858596	0.6667
1.0	0.3004782699105843	0.65
1.0	0.30962778124350177	0.6471
1.0	0.3244957371594926	0.64
1.0	0.326367228113953	0.6364
1.0	0.3394676647951757	0.6316
1.0	0.33957163651486794	0.625
1.0	0.34123518402994385	0.6207
1.0	0.34425036390101893	0.619
1.0	0.3598461218548555	0.6154
1.0	0.36171761280931586	0.6111
1.0	0.36265335828654605	0.6071
1.0	0.3880224578914535	0.6
1.0	0.3936369307548347	0.5833
1.0	0.4017467248908297	0.5789
1.0	0.40257849864836764	0.5714
1.0	0.40372218756498235	0.5625
1.0	0.43127469328342694	0.56
1.0	0.4322104387606571	0.5556
1.0	0.433873986275733	0.5517
1.0	0.4364732792680391	0.55
1.0	0.43834477022249946	0.5455
1.0	0.44395924308588064	0.5417
1.0	0.44915782907049284	0.5385
1.0	0.45009357454772303	0.5357
1.0	0.45019754626741526	0.5333
1.0	0.45477230193387397	0.5294
1.0	0.46153046371386985	0.5263
1.0	0.46236223747140776	0.5238
1.0	0.47275940944063216	0.52
1.0	0.47473487211478477	0.5
1.0	0.47567061759201495	0.4848
1.0	0.5057184445830734	0.48
1.0	0.5092534830526096	0.4762
1.0	0.5205864004990642	0.4737
1.0	0.5343106674984405	0.4706
1.0	0.537117903930131	0.4688
1.0	0.5475150758993553	0.4615
1.0	0.5531295487627366	0.4583
1.0	0.5566645872322729	0.45

(Continued)

Table A1. Continued

TPR	FPR	Threshold
1.0	0.5599916822624246	0.4483
1.0	0.5663339571636515	0.44
1.0	0.566541900603036	0.4333
1.0	0.582761488875026	0.4286
1.0	0.5879600748596382	0.4231
1.0	0.6103139945934706	0.4211
1.0	0.6219588271990019	0.4138
1.0	0.6539821168642129	0.4118
1.0	0.6614680806820544	0.4091
1.0	0.6642753171137451	0.4062
1.0	0.6917238511124973	0.4
1.0	0.6926595965897276	0.3939
1.0	0.699209814930339	0.3929
1.0	0.7044084009149512	0.3846
1.0	0.7060719484300271	0.381
1.0	0.7110625909752547	0.3793
1.0	0.7113745061343315	0.375
1.0	0.7254106882927843	0.3684
1.0	0.7255146600124766	0.3667
1.0	0.7386150966936993	0.3636
1.0	0.7396548138906217	0.36
1.0	0.7424620503223124	0.3571
1.0	0.7561863173216885	0.3529
1.0	0.7658556872530672	0.35
1.0	0.7714701601164483	0.3438
1.0	0.7929923060927427	0.3333
1.0	0.8103555832813475	0.32
1.0	0.8122270742358079	0.3182
1.0	0.8222083593262632	0.3158
1.0	0.8478893740902474	0.3125
1.0	0.8530879600748597	0.3077
1.0	0.8540237055520898	0.303
1.0	0.8574547723019339	0.3
1.0	0.87117903930131	0.2941
1.0	0.8752339363693076	0.2857
1.0	0.8780411728009981	0.2812
1.0	0.9104803493449781	0.28
1.0	0.9170305676855895	0.2727
1.0	0.932626325639426	0.2692
1.0	0.9435433562071117	0.2632

(Continued)

Table A1. Continued

TPR	FPR	Threshold
1.0	0.9463505926388023	0.24
1.0	0.9490538573508006	0.2381
1.0	0.9491578290704928	0.2333
1.0	0.9510293200249532	0.2273
1.0	0.9566437928883343	0.2143
1.0	0.9575795383655645	0.2121
1.0	0.9621542940320232	0.2105
1.0	0.9688084840923269	0.2069
1.0	0.970576003327095	0.2
1.0	0.9732792680390934	0.1905
1.0	0.9788937409024745	0.1818
1.0	0.9798294863797047	0.1786
1.0	0.9877313370763152	0.16
1.0	0.9933458099396965	0.1562
1.0	0.9952173008941568	0.1515
1.0	0.9980245373258474	0.125
1.0	0.9989602828030776	0.1212
1.0	1.0	0.08

A.2 MSSH PRC Values

Table A2. Precision (PREC) and Recall (REC) Yielded by the MSSH, for the Given Threshold, on Images with Metadata, as Presented in Figure 1(b)

PREC	REC	Threshold
0.045454545454545456	1.0	0.08
0.045499701967017685	1.0	0.1212
0.045540419608233074	1.0	0.125
0.045663010967098704	1.0	0.1515
0.045745105872952455	1.0	0.1562
0.04599317131954208	1.0	0.16
0.04634689334142886	1.0	0.1786
0.04638914210473007	1.0	0.1818
0.046644261126387615	1.0	0.1905
0.04676809966302461	1.0	0.2
0.046849427168576104	1.0	0.2069
0.04715815485996705	1.0	0.2105
0.0473727761688043	1.0	0.2121

(Continued)

Table A2. Continued

PREC	REC	Threshold
0.047416916865099905	1.0	0.2143
0.04754983388704319	1.0	0.2188
0.047683498178032274	1.0	0.2273
0.04777302597267133	1.0	0.2333
0.04777800959732944	1.0	0.2381
0.047907949790794976	1.0	0.24
0.04804363788943669	1.0	0.2632
0.048578701739499366	1.0	0.2692
0.049364087087734425	1.0	0.2727
0.049701573521432445	1.0	0.28
0.051443333707738965	1.0	0.2812
0.051599819738621	1.0	0.2857
0.051827543283919884	1.0	0.2941
0.05261344055140724	1.0	0.3
0.052813653136531365	1.0	0.303
0.05286852129747201	1.0	0.3077
0.053175432485777316	1.0	0.3125
0.05474539803968444	1.0	0.3158
0.05538089480048368	1.0	0.3182
0.055501696558410085	1.0	0.32
0.056648113790970935	1.0	0.3333
0.05813658288905814	1.0	0.3438
0.05853783231083844	1.0	0.35
0.05924201267623852	1.0	0.3529
0.06027108830109225	1.0	0.3571
0.06048600105652403	1.0	0.36
0.060565987833906376	1.0	0.3636
0.06159225389994621	1.0	0.3667
0.06160053799596503	1.0	0.3684
0.06273972602739726	1.0	0.375
0.06276552007674387	1.0	0.3793
0.06318112843150779	1.0	0.381
0.06332089036361123	1.0	0.3846
0.06376165947375749	1.0	0.3929
0.0643258426966292	1.0	0.3939
0.06440725636338068	1.0	0.4
0.06689060902585074	1.0	0.4062
0.06715542521994135	1.0	0.4091
0.06787196206283343	1.0	0.4118
0.07111801242236025	1.0	0.4138

(Continued)

Table A2. Continued

PREC	REC	Threshold
0.07237673830594185	1.0	0.4211
0.07492229674464257	1.0	0.4231
0.07554016163615372	1.0	0.4286
0.07753512781445743	1.0	0.4333
0.07756138865368332	1.0	0.44
0.0783709787816564	1.0	0.4483
0.07880247763248452	1.0	0.45
0.0792661820699204	1.0	0.4583
0.08001397624039133	1.0	0.4615
0.0814366998577525	1.0	0.4688
0.0818295515454708	1.0	0.4706
0.0838060384263495	1.0	0.4737
0.08551157580283794	1.0	0.4762
0.08605787298008268	1.0	0.48
0.09099940393403537	1.0	0.4848
0.0911624203821656	1.0	0.5
0.09150849150849151	1.0	0.52
0.09337410805300714	1.0	0.5238
0.09352664896875638	1.0	0.5263
0.09478476821192053	1.0	0.5294
0.09565580618212197	1.0	0.5333
0.09567578859410905	1.0	0.5357
0.09585600669736291	1.0	0.5385
0.09686971235194586	1.0	0.5417
0.09798887462558836	1.0	0.5455
0.09836769759450172	1.0	0.55
0.09889872597711077	1.0	0.5517
0.09924160346695558	1.0	0.5556
0.09943551888840643	1.0	0.56
0.10550564386086156	1.0	0.5625
0.10577367205542725	1.0	0.5714
0.10596945858398889	1.0	0.5789
0.10791705937794534	1.0	0.5833
0.10930787589498807	1.0	0.6
0.11606690319310695	1.0	0.6071
0.11633223266446532	1.0	0.6111
0.11686654758867059	1.0	0.6154
0.12151764393738392	1.0	0.619
0.12245989304812835	1.0	0.6207
0.12298603651987111	1.0	0.625
0.12301907064195541	1.0	0.6316

(Continued)

Table A2. Continued

PREC	REC	Threshold
0.12732832916319156	1.0	0.6364
0.12796870634255378	1.0	0.64
0.13329452852153667	1.0	0.6471
0.13679808841099164	1.0	0.65
0.1378687537627935	1.0	0.6667
0.14131440913298365	1.0	0.68
0.14817211258492397	1.0	0.6842
0.15467747382641	1.0	0.6875
0.15472972972972973	1.0	0.6923
0.15738831615120275	1.0	0.7
0.16380543633762518	1.0	0.7143
0.1642754662840746	1.0	0.7222
0.16534296028880865	1.0	0.7368
0.17179294823705926	1.0	0.75
0.17381404174573056	1.0	0.7619
0.17758821248545947	1.0	0.7778
0.18268847227762267	1.0	0.8
0.1842316975060338	1.0	0.8095
0.18724448078495504	1.0	0.8333
0.1886326194398682	1.0	0.8421
0.1906744379683597	1.0	0.85
0.1909128803668195	1.0	0.8621
0.1921947125472094	1.0	0.8667
0.19227539882451722	1.0	0.875
0.19235615287694247	1.0	0.88
0.19373942470389172	1.0	0.8824
0.19741379310344828	1.0	0.8929
0.19818260493292947	1.0	0.8966
0.19956427015250544	1.0	0.9
0.1996512641673932	1.0	0.9048
0.20078912757562473	1.0	0.92
0.2198751800288046	1.0	0.9231
0.23693740300051733	1.0	0.9286
0.23804573804573806	1.0	0.9375
0.2376495059802392	0.9978165938864629	0.9394
0.2410337552742616	0.9978165938864629	0.9444
0.24218335983041867	0.9978165938864629	0.95
0.24373333333333333	0.9978165938864629	0.9583
0.27105575326215897	0.9978165938864629	0.96
0.288510101010101	0.9978165938864629	0.9688
0.2935131663455363	0.9978165938864629	1.0

Received 12 May 2025; revised 12 May 2025; accepted 7 July 2025