Robust image source identification on modern smartphones

Benjamin Loison

Supervised at Centre Borelli (ENS Paris-Saclay) by:

Quentin Bammey, Rafael Grompone, Pablo Musé and Marina Gardella

July 2, 2024

Contents

1 Introduction

1.1 Motivation

We live in a digitalized world where we face more and more images and videos, and distinguishing truth from falsehood is increasingly difficult. This is notably due to the technological progress of artificial intelligence (AI) in terms of image and video synthesis and deepfakes. This progress involves challenges of unprecedented magnitude regarding misinformation, cybercrime, and privacy violations. Indeed, ill-intentioned parties may use falsified images and videos to manipulate public opinion or even mislead government services. This is particularly dangerous in the current geopolitical context due to the COVID-19 pandemic, wars (Russo-Ukrainian and Israel-Hamas), and democratic elections (Europe, the United States, and the United Kingdom). The forensic analysis of images has emerged as a crucial battlefield for truth, and it becomes imperative to be able to distinguish an authentic image from a falsified or even generated one.

Providing information about the device that captured an image, such as what device model or what device instance took this image, is important for different forensic applications. While the aim of cameras is to capture photos as accurately as possible, in this work, we are interested in the image noise, that is, the difference between the actual scene and the captured photo. In this context, the Photo Response Non-Uniformity (PRNU) is an unintentional image noise component obtained when manufacturing cameras. It consists, for each pixel sensor independently from each other, of incorrectly estimating the actual scene luminance by a constant offset. Hence, during the whole life of the camera, each pixel sensor constantly either overestimates or underestimates the luminance of the actual scene. As a result, the PRNU acts as a unique camera fingerprint. Being able to extract the unique camera fingerprint behind each photo makes it possible to determine which camera took the given photo [LFG06; Gar+23] and detect modifications within the images by retrieving in these regions a different PRNU than those found with another photo from the same camera. Furthermore, for a given camera model, identical hardware architecture and image processing pipeline are involved. As a result, similarly to leveraging the PRNU, it is possible to assign a camera model and an image processing pipeline (specific to a given image editing software, for instance) to photos. These authentication pieces of information are actively used by law enforcement agencies and journalists.

However, modern smartphones implement additional image processing steps in order to further improve the quality of the photos. These extra steps involve merging groups of adjacent pixels in a photo or even merging several shots of the same scene. The performance of current methods leveraging the PRNU is impaired by such heavy image-processing pipelines. As a result, current methods deliver more and more false positives when assigning cameras to photos [IFP21]. Thus, law enforcement agencies and journalists urge scientists to renew existing PRNU extraction methods as they are about to become useless for modern

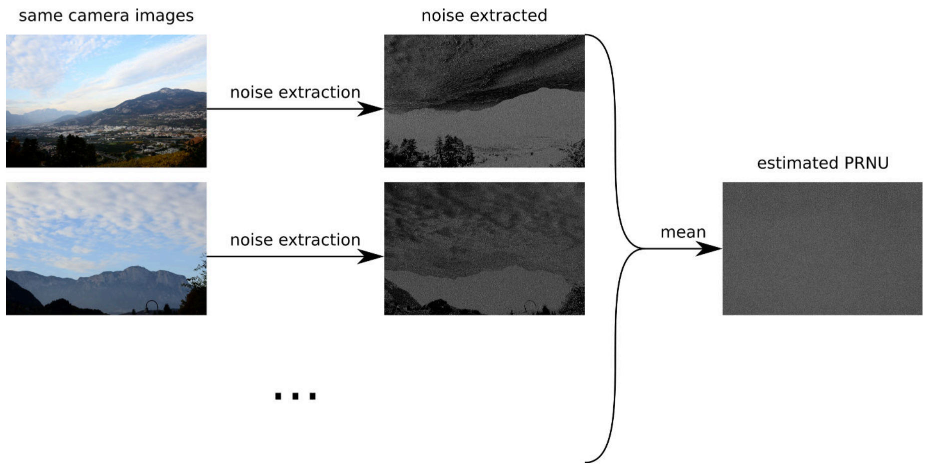


Figure 1: To estimate the PRNU of a given camera, we consider multiple photos of this camera, extract their noise by means of denoising algorithms, and, finally, compute the mean pixelwise

smartphones.

The initial aim of this project was to consider artifacts involved by the heavier image processing pipelines of smartphones leading to false positives for the various forensic tasks leveraging the PRNU. However, even without these post-processing stages, the current state of the art of PRNU analysis suffers from a lack of reproducibility and experiments. This led us to focus on understanding and experimenting with each concept of the estimation and comparison of PRNU in order to continue in the best way this topic in the thesis that will follow this internship, see Section 8.

1.2 Intuition

As depicted in Figure 1, to estimate the PRNU of a given camera, we consider multiple photos of this camera, extract their noise by means of denoising algorithms, and, finally, compute the mean pixel-wise. The theoretical justification is that the extracted noise is the sum of three components: scene residue, random noise, and PRNU. The scene residue is due to the imperfect denoiser used, which interprets part of the actual scene as noise. The random noise, independently from the PRNU, is mostly due to the photonic nature of light. Photons arriving at a pixel sensor follow a Poisson law. The key idea is that averaging on multiple images keeps the PRNU as is and reduces random noise. As we consider different scenes, the scene residue is predominant over the PRNU.

2 Photo Response Non-Uniformity (PRNU)

2.1 Noise definition

The notion of noise in an image comes from the acquisition of images. The first step in acquiring a raw image is to count the number of incident photons on the sensor during exposure time using a sensor. Two different technologies exist for this purpose: charge-coupled device (CCD) and complementary-metal oxide-semiconductor (CMOS). Even if their operating principles differ, their modelization can be made similar [Agu+13]. Both transform incident photons into electronic charges, causing detection devices to generate and store electrons in a potential well. The *noise* in an image thus corresponds to the error from the sensor between the number of photons representing the actual scene and the counted photons.

2.2 Types of noise

There are different types of image noise, divided into two large families.

2.2.1 Pattern noise or non-uniformities

On the one hand, the pattern noises [Jäh10], or non-uniformities, among which the best known is Photo Response Non-Uniformity (PRNU), are related to the position in the image. That is, each pixel, independently from its neighbors, deviates with a constant shift from the correct luminance estimation. These non-uniformities stem from the difference in the accuracy of photon counting in different pixel sensors. These noises can take different forms: an additional count of photons in the dark (DSNU), a slightly low-frequency gradient in the noise variance, periodic noise due to electromagnetic interference in the camera, and finally, a different noise in each pixel due to slight differences in the construction of each sensor, allowing each pixel to be characterized. The non-uniformities, by nature localized, change from one camera to another.

2.2.2 Random noise

On the other hand, the so-called *random* noises represent a variation in the counted photos when capturing an image. They do not depend on the pixel and are generally statistically modeled. Many sources of noise belong to this family: *shot noise* due to the physical nature of the light, *dark current*, which represents a variation around the DSNU seen above, and thermal and electronic reading noises. Under typical capture conditions, *shot noise* dominates. All of these noises in the raw image can be modeled as a Poisson distribution, i.e. a Gaussian noise whose variance linearly depends on the actual luminance.

To illustrate the difference between non-uniformities and random noises, it can be understood that non-uniformities tend to be preserved when several captures of the same scene are averaged, while random noises will be reduced inversely when computing the mean.

3 Raw and processed images

3.1 Usage of each image type

As described in Section 2.1, cameras count incident photons at each pixel sensor during the exposure time and produce an unprocessed (or minimally processed) image qualified as raw. In practice, however, the observed images are not raw but go through a traditional processing pipeline to become visually intelligible, a simplified version of which can be seen in Figure 2.Usually, photographers prefer raw images to precisely process their photos the way they want, while smartphone end-users prefer already processed images that are visually intelligible and compressed to be ready to be shared. As a result, separate cameras often provide both raw and processed images, while smartphones usually only provide processed images.

Even when smartphone cameras provide raw images, note that these images are not really raw. Indeed, the Image Signal Processing (ISP) pipeline of smartphones usually contains trade-secret steps, including pixel binning and burst images. The registered burst images and binned pixels are then aggregated to form a quasi-raw image that is exempt from further post-processing, such as color and optical corrections, but not fully exempt from processing.

3.2 Processing steps

3.2.1 Raw image capture

As is, raw images would be grayscale. To add the color information, a Color Filter Array (CFA) is used in the camera to filter the incident photons by color channel. To do so, the color filter array, consisting of a mosaic of tiny color filters, is placed over the pixel sensors. A well-known mosaic is the Bayer pattern, depicted in Figure 3. It is made of a repeated 2×2 matrix of pixel sensors with two green channels in a diagonal and one red and one blue channel in the other diagonal. Hence it is also named BGGR, GBRG, GRBG, or RGGB. As a result, each image pixel at this stage has an intensity of either red, green, or blue. As there is a majority of green pixel sensors, the image at this stage looks greenish. This visual aspect will be corrected at the color and optical corrections step described in Section 3.2.3.

3.2.2 Demosaicing

Demosaicing consists of interpolating the missing color values at each pixel to make each pixel have an intensity in the 3 color channels, Red, Green, and Blue (RGB), instead of keeping each pixel at an intensity in either red, green, or blue. This operation keeps the original image resolution. In the raw image, coming out directly from the sensors, many models exist for these noise sources. Demosaicing uses neighbor pixels to interpolate the missing color channel values and introduces, in particular, a spatial correlation in noise; that is, the final pixel noise depends on the noise of its neighbor pixels. However, this incomplete

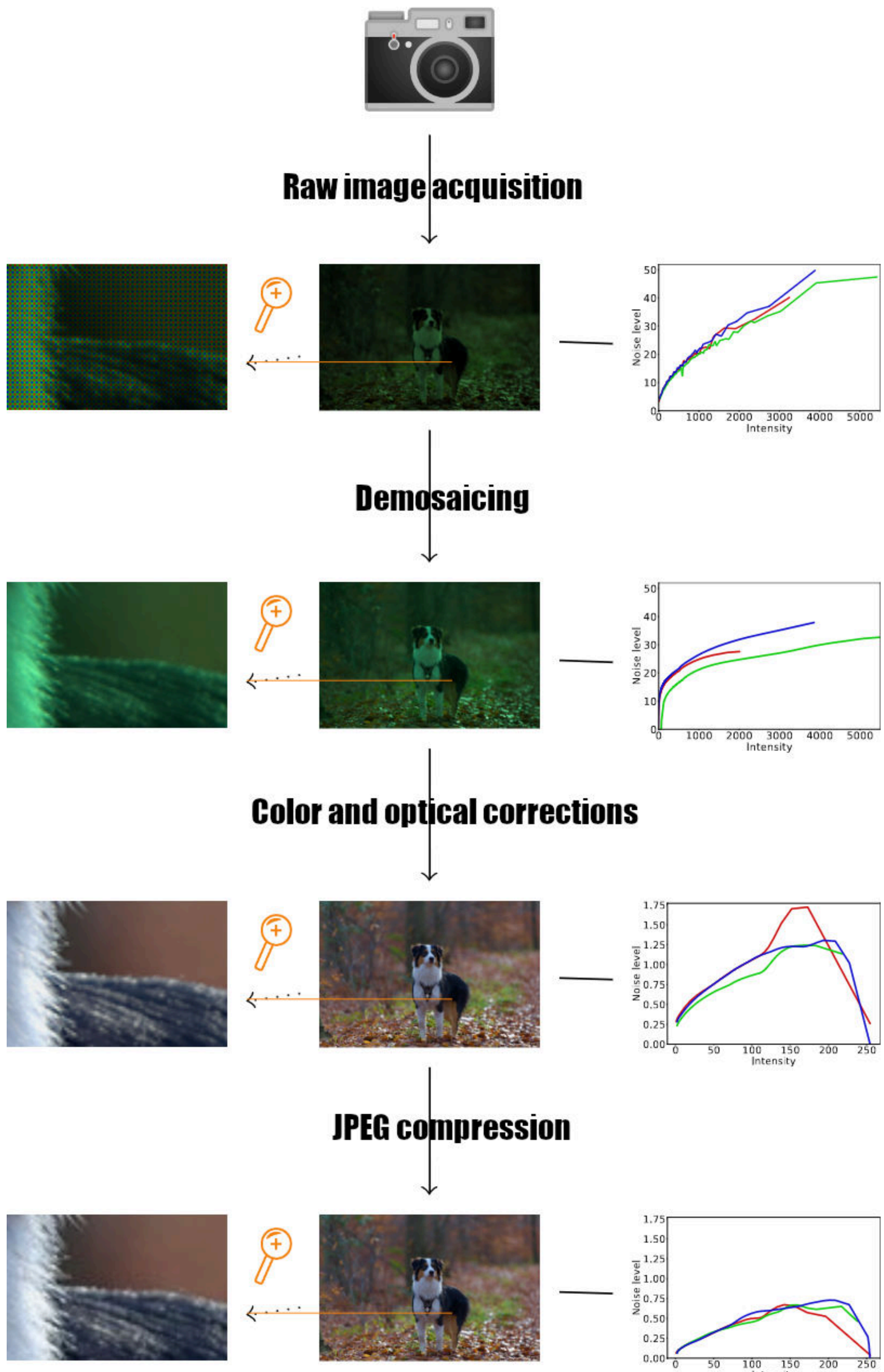
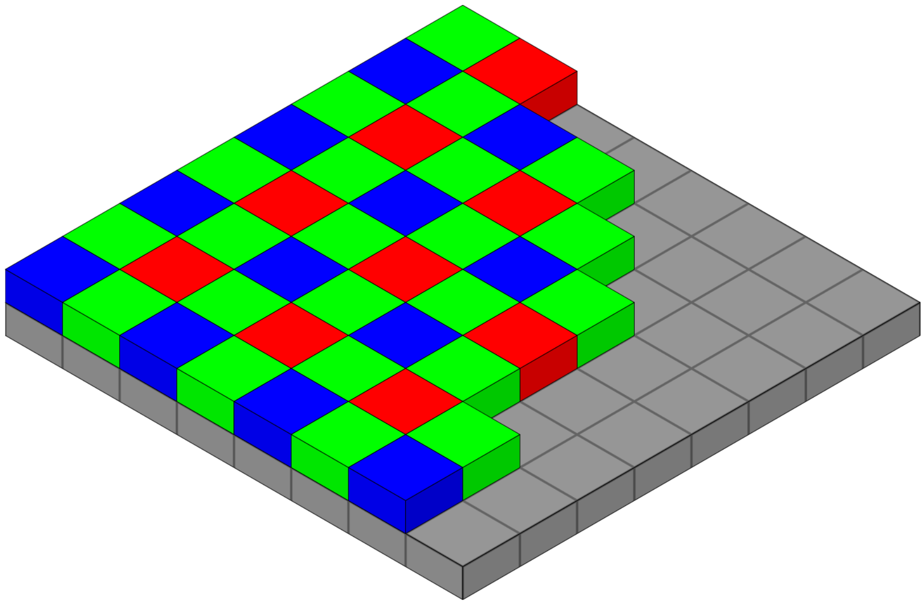


Figure 2: Simplified image processing pipeline, and noise curve associated with each step [Bay].



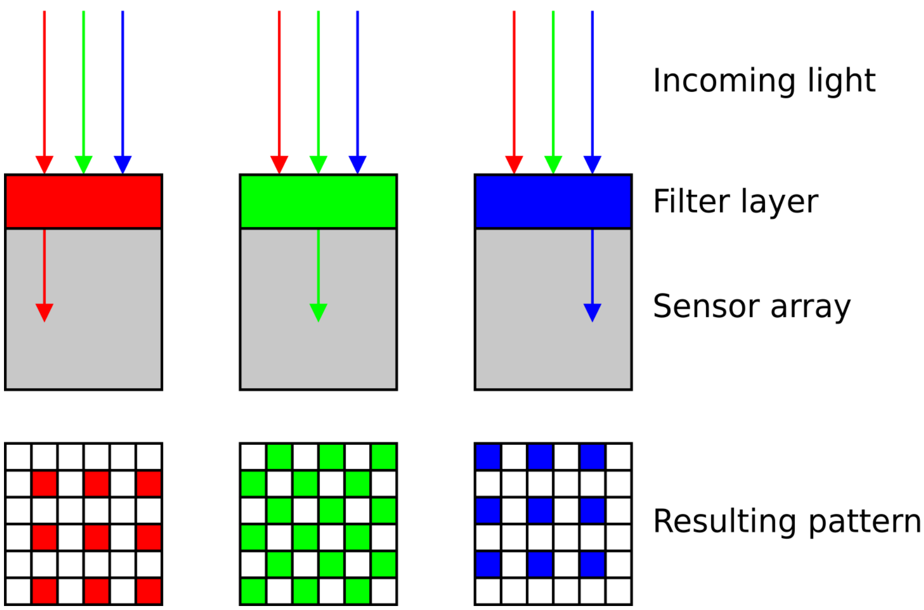


Figure 3: To add the color information, a Color Filter Array (CFA) is used in the camera to filter the incident photons by color channel. To do so, the color filter array consists of a mosaic of tiny color filters placed over the pixel sensors. A well-known mosaic, represented here, is the Bayer pattern.

model was completed only in 2020 [JFM20]. To correctly estimate the PRNU, the spatial correlation has to be taken into account, as the theoretical PRNU fixed pattern is intuitively blurred due to spatial averaging.

3.2.3 Color and optical correction

Due to the color filter array used during the raw image capture, described in Section 3.2.1, after demosaicing, the green color channel is over-represented in comparison with the other color channels. A color correction is applied by an algorithm to make colors accurately represent the captured scene.

The camera lens introduces notably two artifacts: optical distortion and vignetting. The former, due to the lens shape, gives a bumpy aspect. The latter is a decrease of the image brightness towards the periphery compared to the image center, see the corners of Figure 12. Both artifacts are corrected thanks to an algorithm [Maı17].

3.2.4 Compression

While raw images use 12 or 14 bits per pixel, processed images use 8 bits per color channel for each pixel. Hence, each pixel uses 24 bits. As each image is a matrix of thousands by thousands of pixels, storing and sending images efficiently is crucial. To do so, images are compressed either losslessly, that is, without loss of information as with the PNG compression format, or with a lossy compression algorithm such as JPEG. Note that for both compression schemes, there is a quantization. While PNG compression allows the quantized data to be perfectly reconstructed from the compressed data, JPEG compression adds even more spatial correlation, making current PRNU extraction methods no more theoretically justified. The spatial correlation introduced by JPEG compression is due to the quantization, the split and the compression of the image into blocks of 8x8 pixels up to a given precision in the discrete cosine transform used. Similarly to demosaicing introducing spatial correlation, as described in Section 3.2.2, if many studies highlight the artifacts created during JPEG compression of an image, it was only in 2022 that a partial model was created to understand the impact of compression on the pre-existing noise in the image [Gar+22].

3.3 Additional smartphone processing

3.3.1 Additional smartphone processing operations

In addition to the traditional image pipeline presented in Section 3.2, modern smartphones proceed to additional image processing operations. The additional operations modern smartphones proceed to involve the non-unique artifacts we will describe in Section 3.3.2. The additional operations of modern smartphones are presented below:

- Pixel binning: as shown in Section 3.2.1, images are typically sampled using a matrix of colored filters, commonly the Bayer [Bay] pattern. Now, most sensors sample several adjacent pixels in the same color, and can, depending on the image, join these pixels in one [JH12].
- Burst mode and multiple sensors: cameras embedded in mobile phones usually take several shots from the same scene, with the same camera sensor or different a different one. These images are then merged into a single one of superior quality [LFE23; Lec+22].
- Models of image restoration by diffusion begin to appear, both for denoising, super-resolution and overall improvement of images [Fei+23; Gao+23; HJA20; Li+22].

3.3.2 Non-unique artifacts issue

As mentionned in the previous Sections 1.1 and 3.3.1, modern smartphones improve the quality of photos by adding image processing steps, for instance, merging neighboring pixels and photos. In addition to the already well-known demosaicing and compression steps, these modern smartphone operations increase the spatial correlation making accurate PRNU extraction no more theoretically justified. As multiple cameras of a given manufacturer model use an identical heavy image processing pipeline, the precise spatial correlation implied by this image processing pipeline blurs the PRNU the same way. As a result, each camera is less distinguishable from each other. The specific spatial correlation introduced by a given heavy image processing pipeline is named non-unique artifacts, as all cameras having this heavy image processing pipeline have these artifacts. As a result, current methods have more and more false positives when assigning cameras [IFP21]. Users of these PRNU extraction methods urge scientists to renew these existing methods as they are about to become useless for modern smartphones.

4 Photo Response Non-Uniformity (PRNU) extraction

This section completes Section 1.2 giving a first intuition of the PRNU extraction. In particular, PRNU-based techniques are used by law enforcement agencies to determine whether one or more ordinary photos, called non-flat-field, come from a given camera. An example of non-flat-field photo we considered for experiments detailed in Section 6 is provided in Figure 5. Since this scientific evidence can be used in a trial, a very low rate of misallocation of the source camera must be guaranteed. We will present how the PRNU of a camera can be extracted from one or more photos. The more photos we consider, the greater our confidence in source camera assignment. In the context of justice, law enforcement authorities may have access to cameras suspected of having taken the given photos. In order to determine which of these cameras, if any, has taken

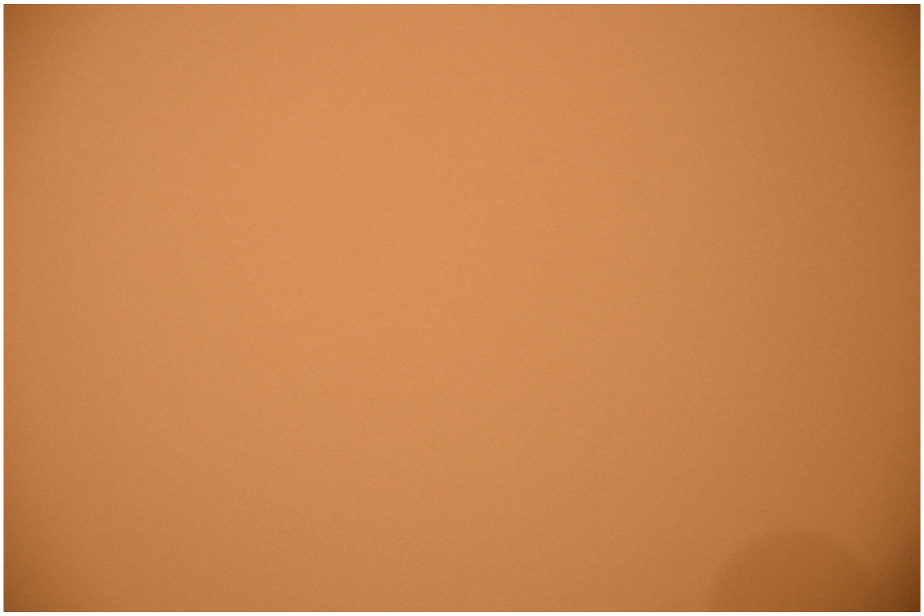


Figure 4: Example of RAISE dataset flat-field photo with Nikon D7000 camera [Dan+15]. The scene looks like an orangeish wall.

the given photos, we extract and compare the PRNU of the photos considered and the one of the camera. To get an optimal extraction of the PRNU of the camera, evenly illuminated photos, called flat-field photos, are captured under controlled conditions with the camera. An example of flat-field photo we considered for experiments detailed in Section 6 is provided in Figure 4. In this way, the scene content leaked to the noise residues described in Section 1.2 are minimized.

4.1 Methods

Depending on the type of photos considered, that is flat-field or not, more or less methods for extracting the PRNU can be applied.

4.1.1 Trivial mean flat-field same scene photos

When considering multiple flat-field photos of the same scene taken by a given camera, a trivial method can be exploited to extract the noise on these photos, hence extract the camera PRNU. Indeed, for such a specific case, the noise of images can be obtained by subtracting the average of the images from the images. This is because the average of the images acts like a denoised version of the images, as averaging the random noise of the considered images makes the mean random noise converge to zero. Note that for this method to work

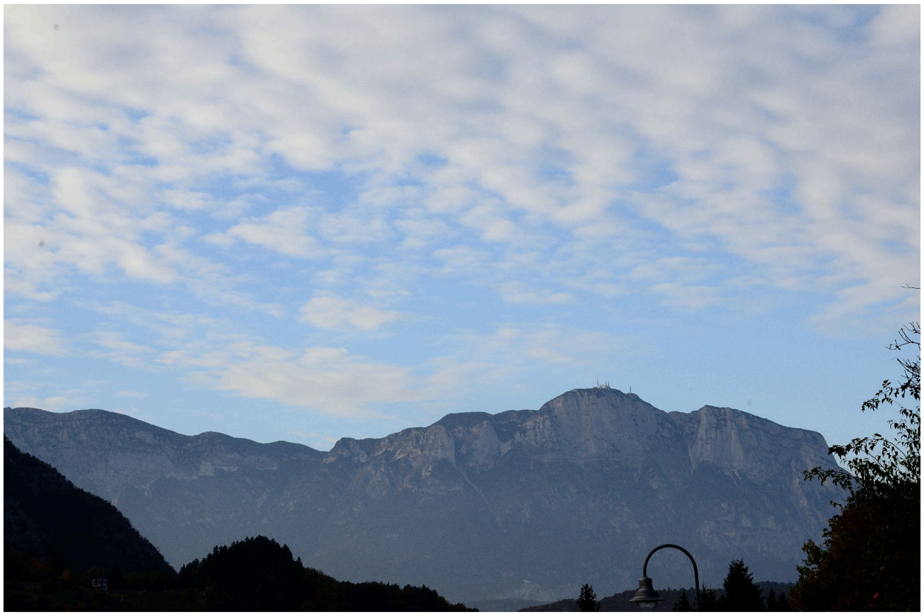


Figure 5: Example of RAISE dataset non-flat-field photo for Nikon D7000 camera.

correctly the scene captured has to be exactly the same. So the camera should not move during the capture of the multiple images (by laying the camera on a surface or by using a tripod) and the scene should not change (attention should also be paid to possible change in illumination and shadows, if any). When the camera is available to take flat-field images, this method can thus be used as a ground truth, as it is trivial, notably as it does not use any denoiser. Compared to the following more general method using a denoiser, the denoiser used here consists in subtracting the average of the considered photos from each photo.

For multiple flat-field photos of the same scene taken by a given camera we can also extract the noise by subtracting from each image the same image blurred with a Gaussian kernel.

4.1.2 More general method without same-scene photos

The previous trivial mean method does not perform well for photos considering different scenes as the average of the considered images would not result in a denoised version of each image as they represent different scenes. To extract the noise in each image we use a denoiser. The main issue with denoisers is that they are not perfect, such that some scene residues are still present in the extracted noise, and denoisers may even involve artifacts.

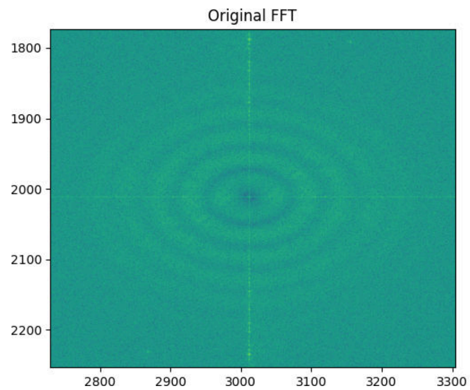


Figure 6: PRNU in the Fourier domain with attenuated axes.

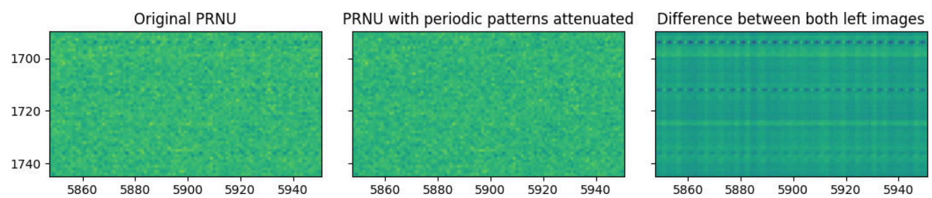


Figure 7: PRNU in the image domain with attenuated axes. We clearly notice the removal of the dotted lines.

4.2 PRNU post-processing

In the previous sections, we focused on the extracted noise of the images, but the actual unique fingerprint of the cameras consists in their PRNUs. To make the extracted noise closer to the camera PRNU, multiple post-processing operations can be applied [Che+08]:

- Subtract from each noise residue pixel the average of its column, then subtract the average of its row
- Set pixels of axes in the Fourier domain to the average of its local neighbor pixels, such that periodic patterns in the domain image are removed, see Figures 6 and 7.

Similarly to [DSM07], we found out in the estimated PRNU of RAISE Nikon D7000 and Rafael Grompone's flat-field photos what looks like dust circle artifacts, see respectively the Figures 8 and 9. While these dust circle artifacts may be used to uniquely identify cameras, they are probably less constant along time than the PRNU. The circle artifacts in the image domain also result in centered circles in the Fourier domain with a clear radial profile easing processing them, see Figures 6 and 10.

Figure 11 summarizes the various typical steps for estimating a camera PRNU. For each step multiple approaches exist. There are multiple denois-

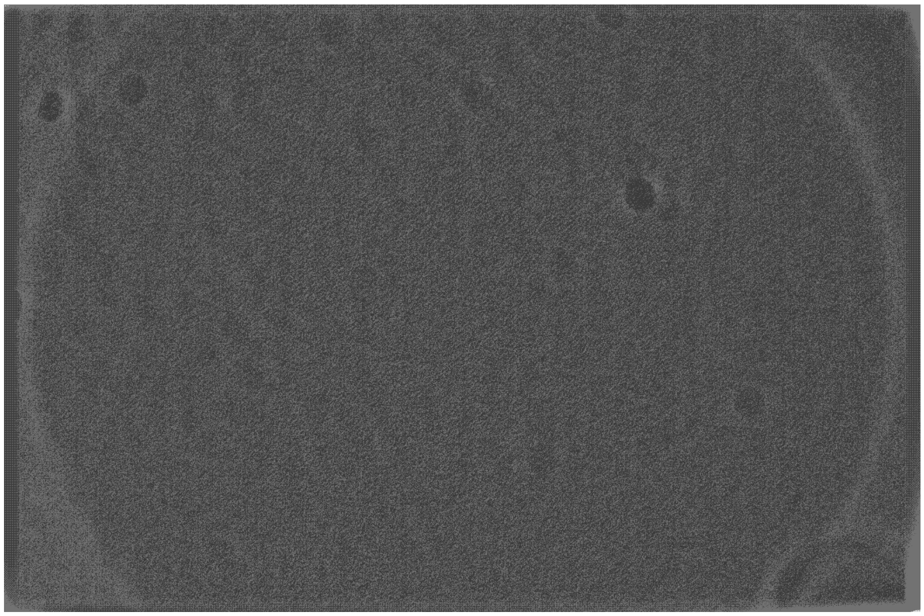


Figure 8: RAISE Nikon D7000 camera estimated PRNU using a bilateral denoiser. We notice circle artifacts, which might be dust.

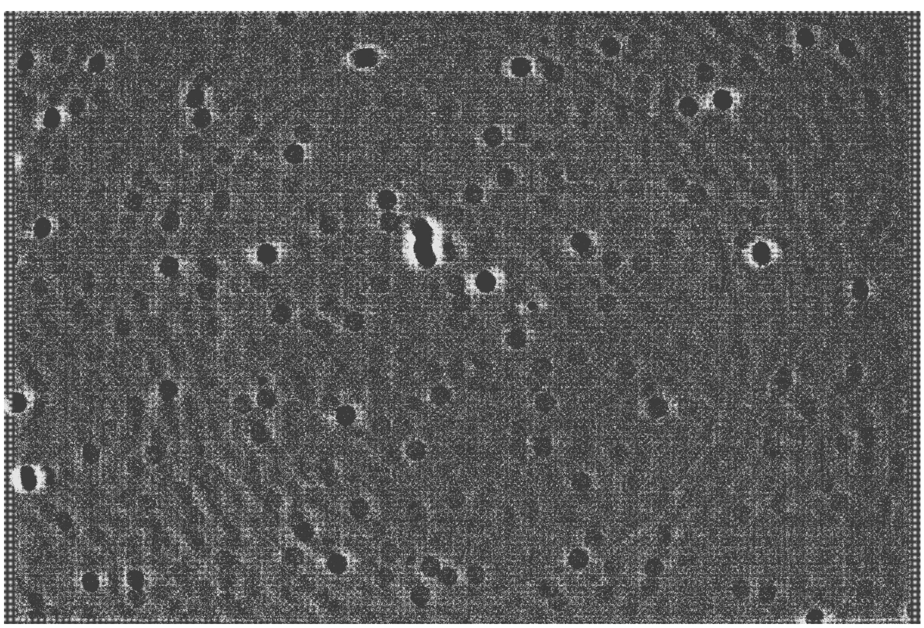


Figure 9: Rafael Grompone's camera estimated PRNU using a bilateral denoiser. We clearly notice circle artifacts, which might be dust.

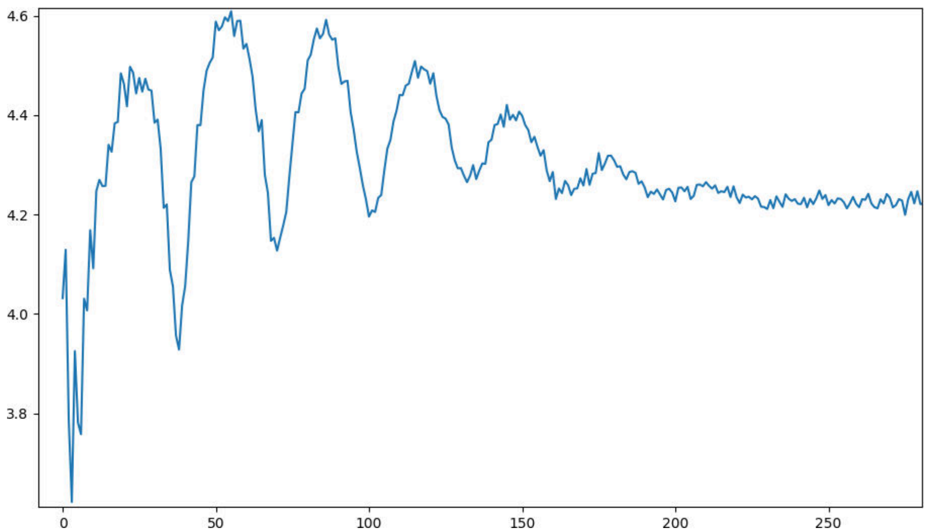


Figure 10: Radial profile in the Fourier domain. The x-axis is the distance to the circle center and the y-axis is the average of the pixels in the Fourier domain at this distance from the circle center.

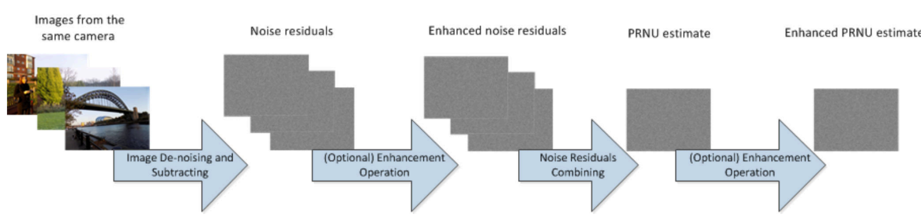


Figure 11: A diagram of the typical PRNU estimation procedure [AK17].

ers, methods of combining noise residuals and PRNU estimate enhancement operations.

5 PRNU comparison

Once we have estimated the PRNUs for two sets of images, we would like to compare them to know if both sets of images were acquired with the same camera. As PRNUs are images of noise, technically the aim is to evaluate how close both images of noise are. Note that the comparison between PRNUs has to be done carefully as such a result could be used as evidence in court.

In the first place, distance metrics such as Root Mean Square (RMS) and Pearson correlation coefficient can be considered. The Pearson correlation coefficient is a correlation metric between -1 and 1 where:

- 1 means that both images are perfectly correlated

- 0 means that both images are not correlated
- -1 means that both images are perfectly inversely correlated

The Pearson correlation coefficient presents an important limitation: with the presence of non-unique artifacts, its value raises and produces false positives [Gar+23]. To address this issue, [Gol08] proposed to use the peak-to-correlation energy (PCE), which suppresses the effect of periodic pattern. To preserve the information about the sign of the correlation, a signed version of the PCE can be used instead. The sPCE metric is the preferred test statistic in source camera identification applications.

Note that in a use case scenario the image orientation is unknown, so to *correctly* test the presence of the PRNU, the sPCE is computed for each possible rotation and the highest computed sPCE is kept.

To give an order of magnitude in [GFF09], the authors tested 1,024,050 mismatching images and got a maximum sPCE value of 57. Therefore, they conclude that, depending on the JPEG quality of the images, a threshold of 60 guarantees a false alarm rate smaller than 10^{-6}.

6 Experiments

6.1 Introduction

As described in Section 1.1, we had a step-by-step approach regarding the experiments, starting with estimating, then comparing PRNUs in a very controlled environment to finally compare them in a practical usage environment. In fact, the very first images we considered were ones we synthetically generated ourselves, see Section 6.3.1.

In addition to the generated images we started with, we used two sources of images:

- RAISE proposes non-flat-field photos for 3 Nikon cameras and flat-field photos for one of these cameras being the Nikon D7000 [Dan+15]. In addition to the latter camera we focused on the Nikon D90 being the second camera having taken the most photos. These flat-field photos captured an orangeish wall, see Figure 4. This camera generated both raw images (.nef) and processed images (.tif).
- Rafael Grompone's camera, a SONY ILCE-6000. This camera generated both raw images (.arw) and processed images (.jpg). We first worked with photos of a white wall and dark sky but then images of a bright cloudless sky, see Figure 12.

Note that the experiments were coded in Python and are publicly available at [Loi24a; Loi24b]. Depending on the denoiser used, the denoising time varies significantly. The trivial mean flat-field same scene method when applicable and the wavelet denoiser for other cases can help to prototype but using bilateral

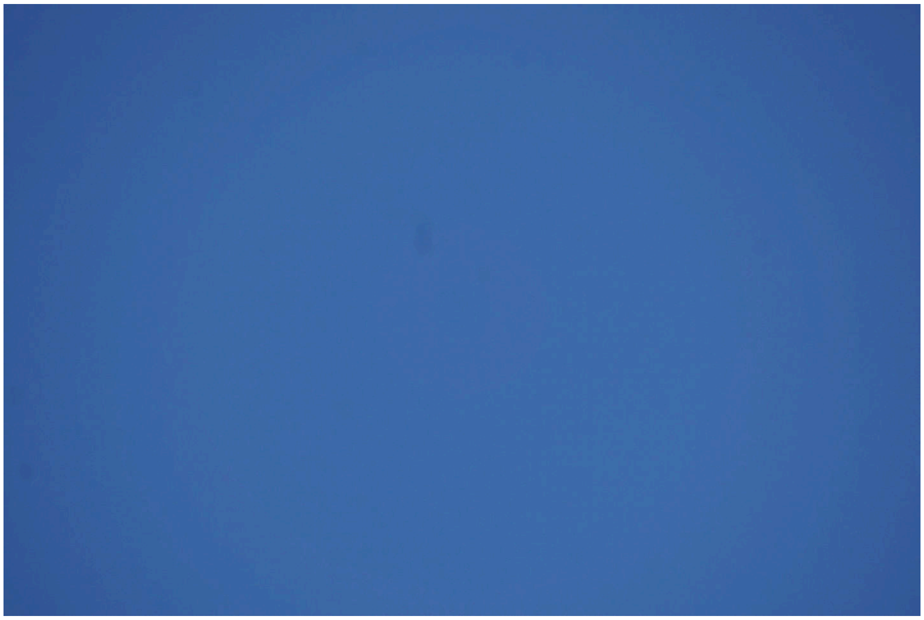


Figure 12: Example of Rafael Grompone flat-field photo taken with a SONY ILCE-6000 camera. The scene is a cloudless sky.

denoiser seems more correct. Indeed, the bilateral denoiser has less column and row artifacts, and it makes unique artifacts, such as dust circles described in Section 4.2, more significant than when using the unbiased trivial mean flat-field method. Similarly, concerning comparing PRNUs, sPCE computation takes significantly more time than other metrics.While considering only centered crops can provide interesting results, our parallelized implementation is able to execute in a few minutes our experiments involving using the bilateral denoiser and sPCE for 200 images on an AMD EPYC 7371 processor. To read raw images, we use the rawpy library. To the best of our knowledge, we have to serialize raw images into Numpy arrays to speed up their successive loads when using crops, thanks to mapping RAM memory and disk to avoid loading the whole images. This is especially true when all experiment (crops of) images do not fit into memory. Alternatively, when all (crops of) images do not fit into memory, computing directly the mean of n images with $mean = \frac{images}{n}$ may not be possible. However, it is possible to compute the mean iteratively for i between 1 and n, with:

$$mean_i = \frac{mean_{i-1} * (i-1) + image_i}{i},$$

and $mean_0$ the image made of zeros and with the same resolution as other images. With this computation method $mean = mean_n$. That way by proceeding one image at a time, we have a memory complexity of a single image plus an integer to keep track of how many images composed the iterative mean, instead of n images.

6.2 Denoisers

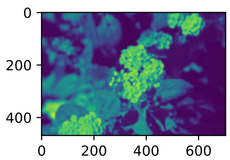
First we proceeded with the wavelet denoiser [CYV00] but as it may involve significant artifacts we moved to notably bilateral [TM98] and low pass denoisers. While we use skimage library for the wavelet and bilateral denoisers, concerning the low pass denoiser we implemented it on our own.

The low pass denoiser consists in applying a Fourier transform to the image we want to denoise. In this way, we have a frequential view of the image, so computing the convolution with a unit intensity centered circle and applying an inverse Fourier transform removes the high frequency components of the image. As previously, to only obtain the noise instead of the denoised image, we can subtract the denoised image from the image itself. Similarly to the low pass denoiser implementation, to extract the image noise, we can just do the inverse, that is a high pass filter, consisting in computing in the Fourier domain the correlation of the image with a same size unit intensity image subtracted by a unit intensity centered circle. Note that to avoid introducing a frequential intensity artifact on the denoised image or image noise, a centered circle with a decreasing intensity has to be used instead of a constant unit intensity.

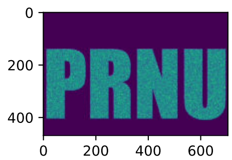
6.3 Types of images

6.3.1 Generated images

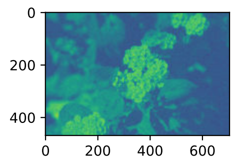
One of the first experiments we conducted consisted in estimating an artificially generated PRNU. To do so, we considered a synthetic, noiseless, non-flat-field image, see Figure 13a, to which we added Gaussian noise, see Figure 13c, and the PRNU, see Figures 13b and 13d. This experiment is depicted in Figure 13. The artificial PRNU used for this experiment is an image with "PRNU" written using Gaussian noise, see Figure 13b. As expected, we notice in Figure 13e, 13f and 13g that the PRNU gets more and more significant when averaging more and more modified photos. To make the denoiser work harder we have added noise to the noiseless images and wrote "PRNU" using Gaussian noise. We first considered images processed such that it is assumed that they do not contain noise. To be closer to realistic PRNUs, we considered writing PRNU not as either 0, where "PRNU" is not written, or 1, where "PRNU" is written, but instead for the latter as a Gaussian noise, see *Actual Gaussian noised PRNU*. To avoid a trivial detection of where there is Gaussian noise, we first added to the noiseless image some Gaussian noise with a wider variance than the variance used to write "PRNU". Visually we clearly notice in Figure 13c the quite strong Gaussian noise compared to the 13d where we do not even barely notice the text "PRNU". The more photos we average to estimate the PRNU, the more it is estimated correctly. Starting at 96 photos, we start visually to be able to read by naked eye the text "PRNU". It is interesting to note that the Root-Mean-Square (RMS) distance between the estimated PRNU and the actual PRNU



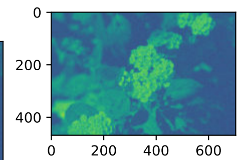
(a) First image without noise



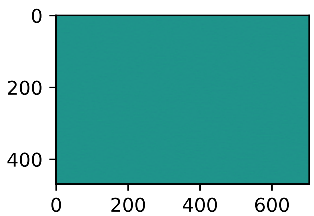
(b) Actual Gaussian noised PRNU



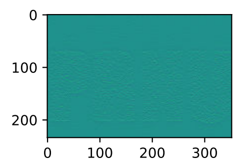
(c) First image with Gaussian noise



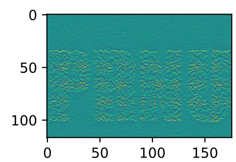
(d) First image with Gaussian noise and PRNU



(e) Number of images: 16
RMS with actual PRNU: 0.9425



(f) Number of images: 96
RMS with actual PRNU: 0.9323



(g) Number of images: 320
RMS with actual PRNU: 0.9455

Figure 13: PRNU estimation with different number of images having Gaussian noise and Gaussian noised PRNU

does not decrease when we consider more and more photos while we visually notice the text “PRNU” increasingly better. Note that the resolution of the estimated PRNU image changes as we split the original photos into more pieces to match the wanted number of photos to better estimate the PRNU.

6.3.2 Flat-field images

Contrary to the previous case, we consider here photos with natural noise including PRNU. Hence, evaluating the correctness of the PRNU estimation looks less clear, as we do not have any known PRNU ground truth for the considered camera.

If we consider multiple photos for each camera, then we can estimate the PRNU for each camera. In that way we can compute an accuracy in assigning cameras to new photos by guessing that the camera having the lowest distance between its estimated PRNU and the one of the considered photo is most probably the camera having taken this photo.

If we consider multiple photos for a single camera, then we cannot evaluate our approach by measuring the accuracy in camera attribution, as there is a single camera. However, we can split the photos into two groups and estimate the PRNUs among each group. The more photos we consider, the more we would expect both PRNUs to be closer to each other. However, we have to verify that both estimated PRNUs are not trivial purely black images.

With photos of a single camera, we can artificially generate multiple camera-like photos. Indeed, thanks to the independence of the PRNU from a pixel to the other, we can for instance split with a photo vertically into two equal parts to get two photos from each photo. In this case, the left-half photos and the right-half photos behave like two different cameras. Being able to predict which *half* camera took a given half photo, shows that we are able to leverage the PRNU location. On the Figure 14, we notice that for RAISE flat-field TIF photos, we are able to perfectly predict whether the provided photo is the left or right half of the camera. This experiment consists in considering 100 photos for each camera and computing the sPCE within random pairs of photos where both photos are from the same camera or from two different ones. That way we have 100 random pairs of photos from two different cameras and 100 random pairs of photos from the same camera. As expected, the sPCE between pairs of images:

- of two different cameras is near zero, as both PRNU are independent. In fact the sPCE is near 10, as we probably have not post-processed the PRNU that is just the extracted noise. So the estimated PRNU probably contains common artifacts due to the identical processing performed by the camera on the left and right halves.

- of the same camera is high, as both PRNU are identical. It is also interesting to note that the left half has a higher sPCE than the right half, as the latter is darker notably in terms of vignetting and with the bottom right shadow possibly being the camera operator's head.

This experiment prediction is perfect, as no sPCEs of pairs of images of the same camera is lower than the sPCE of pairs of images of two different cameras. In addition, this perfect prediction is quite reliable in the sense that there is quite a significant gap of about 40 between the sPCEs of pairs of images of the same camera and the ones of two different cameras.

An additional interesting remark is that being able to predict whether half of a photo is from the left or the right half of the camera means that we can generalize to some extent the prediction of the location of a photo crop. In other words, if a part of a photo is falsified, then the PRNU might be altered and we might be able to detect such a forgery [Che+08].

6.3.3 Non-flat-field images

Figure 15 shows the results of the same experiment conducted in a less controlled environment that is with left and right halves photos of flat-field for the same RAISE Nikon D7000 camera. These results are similar to the previous experiment with flat-field photos except that the source camera prediction is not perfect anymore. Indeed, there are two incorrect predictions and, furthermore, predictions are less reliable, as there is no more gap between distributions. Such degradation of performances can be expected with non-flat-field photos as they are taken in a less controlled environment.

As RAISE proposes multiple non-flat-field photos per camera, we ran the same experiment as previously but with two different camera models, Nikon

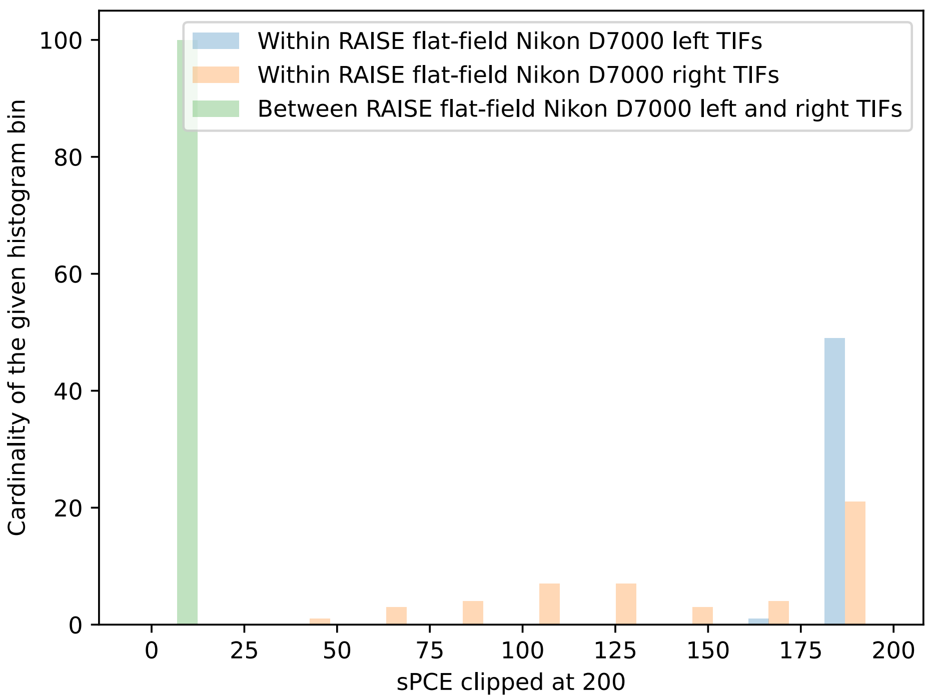


Figure 14: sPCEs of pairs of flat-field images within and across left and right halves of RAISE Nikon D7000 camera

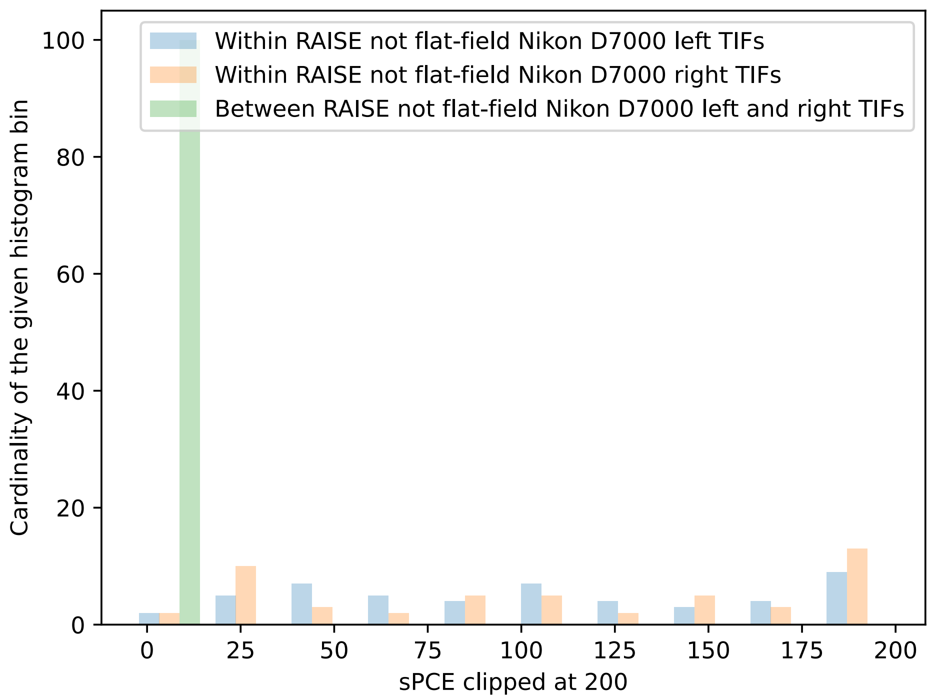


Figure 15: sPCEs of pairs of non-flat-field images within and across left and right halves of RAISE Nikon D7000 camera

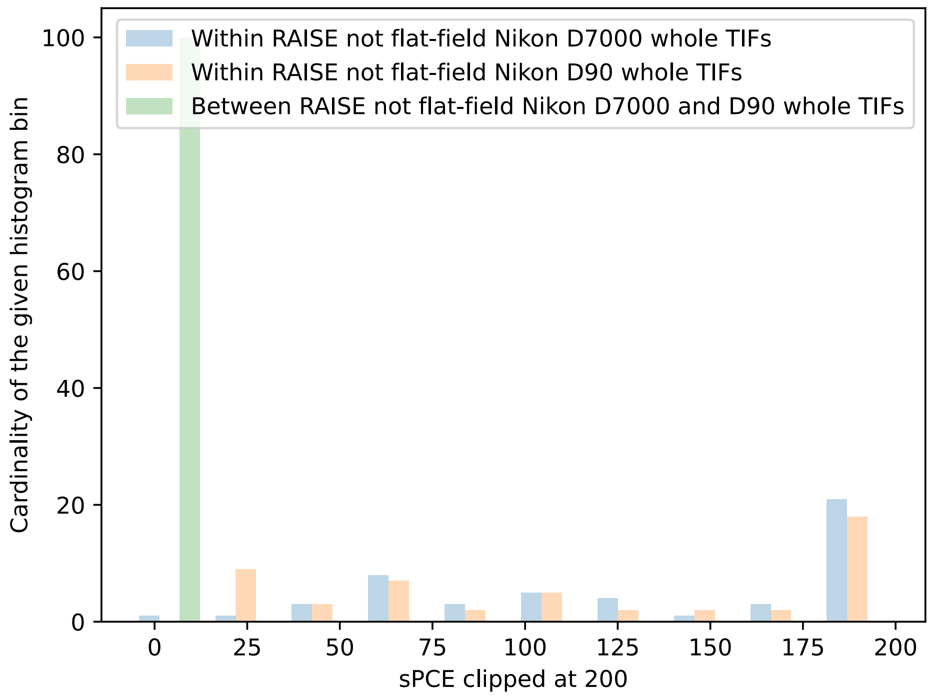


Figure 16: sPCEs of pairs of non-flat-field images within and across RAISE Nikon D7000 and Nikon D90 cameras

D7000 and Nikon D90, from the same manufacturer, Nikon, instead of considering the left and right halves of the same Nikon D7000 camera. The results of this experiment are depicted in Figure 16. They are very similar to the previous experiment results. These results show that the PRNU, as a unique camera fingerprint, is effectively estimable.

Figure 17 is a zoom on the 100x100 centered crop of the estimated PRNU from Figure 1, it shows that the camera estimated PRNU is globally random.

7 Conclusion

Although we have not had the opportunity to investigate the issue of PRNU with smartphones yet, we have established a public step-by-step reliable PRNU estimation and comparison method that we evaluated with generated, practical raw and processed photos.The results of these experiments show with reliability that the PRNU, as a unique camera fingerprint, is indeed estimable and can be localized to possibly detect forgeries within the considered images.

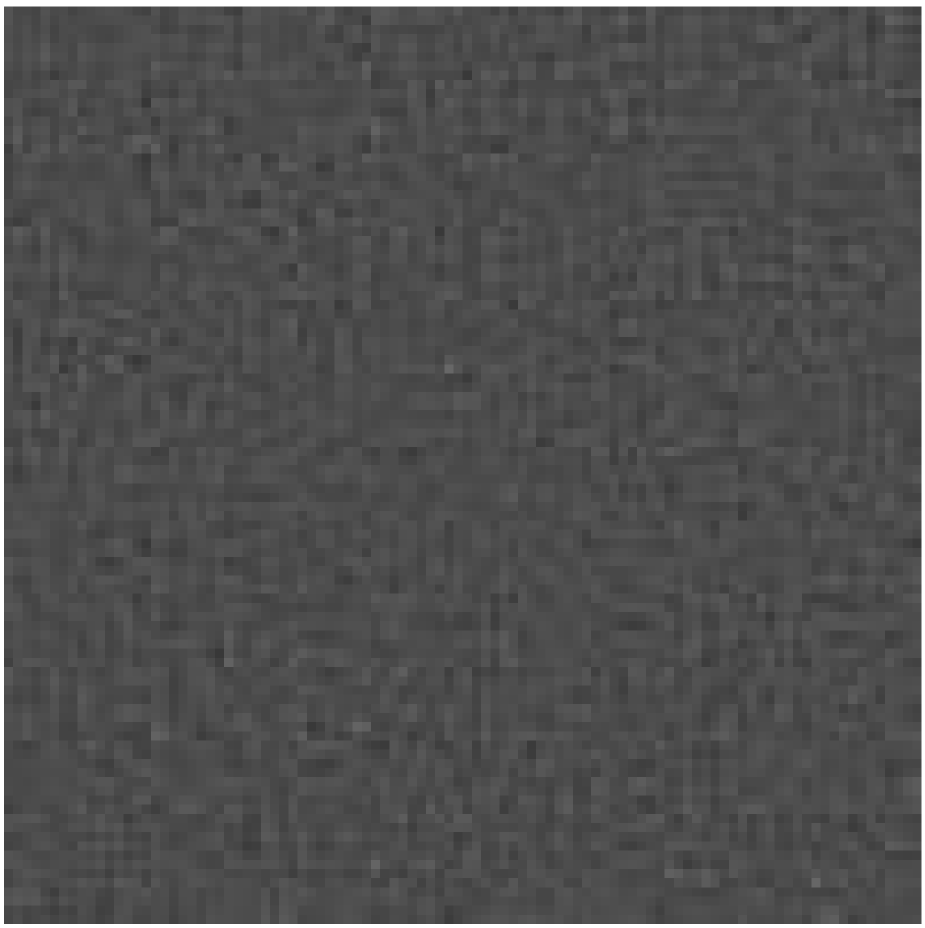


Figure 17: 100x100 centered crop of RAISE Nikon D7000 camera estimated PRNU with 50 non-flat-field TIF photos.

8 Meta-information

This work is intended to continue next year into a thesis entitled *Murmurs of images: Forensic analysis of image noise patterns and structures for the detection of falsified images*, funded by the “Agence Ministérielle pour l’IA de Défense” (AMIAD) located at the École Polytechnique. To help spread this reproducible research on estimating and comparing PRNUs, an IPOL MLBriefs article [Loi+24a] with an online demo [Loi+24b] is planned to be submitted. To iterate quickly with experiments we achieved to leverage correctly, that is with our wanted up-to-date software programs, the possible parallelism of one 32 threads IPOL computation server thanks to Nix.

References

[Agu+13] Cecilia Aguerrebere et al. *Study of the digital camera acquisition process and statistical modeling of the sensor raw data*. Tech. rep. Aug. 2013. URL: https://hal.archives-ouvertes.fr/hal-00733538.

[AK17] Mustafa Al-Ani and Fouad Khelifi. "On the SPN Estimation in Image Forensics: A Systematic Empirical Evaluation". In: *IEEE Transactions on Information Forensics and Security* 12.5 (2017), pp. 1067–1081. DOI: 10.1109/TIFS.2016.2640938.

[Bay] Bryce E. Bayer. *Color imaging array*. US3971065A.

[Che+08] Mo Chen et al. "Determining Image Origin and Integrity Using Sensor Noise". In: *IEEE Transactions on Information Forensics and Security* 3.1 (2008), pp. 74–90. DOI: 10.1109/TIFS.2007.916285.

[CYV00] S.G. Chang, Bin Yu, and M. Vetterli. "Adaptive wavelet thresholding for image denoising and compression". In: *IEEE Transactions on Image Processing* 9.9 (2000), pp. 1532–1546. DOI: 10.1109/83.862633.

[Dan+15] Duc-Tien Dang-Nguyen et al. "RAISE: a raw images dataset for digital image forensics". In: *Proceedings of the 6th ACM Multimedia Systems Conference*. MMSys '15. Portland, Oregon: Association for Computing Machinery, 2015, pp. 219–224. ISBN: 9781450333511. DOI: 10.1145/2713168.2713194. URL: https://doi.org/10.1145/2713168.2713194.

[DSM07] A. Emir Dirik, Husrev T. Sencar, and Nasir Memon. "Source Camera Identification Based on Sensor Dust Characteristics". In: *2007 IEEE Workshop on Signal Processing Applications for Public Security and Forensics*. 2007, pp. 1–6. DOI: 10.1109/IEEECONF12259.2007.4218950.

[Fei+23] Ben Fei et al. "Generative Diffusion Prior for Unified Image Restoration and Enhancement". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 9935–9946.

[Gao+23] Sicheng Gao et al. "Implicit diffusion models for continuous super-resolution". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 10021–10030.

[Gar+22] Marina Gardella et al. "The Impact of JPEG Compression on Prior Image Noise". In: *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2022, pp. 2689–2693.

[Gar+23] Marina Gardella et al. "PRNU-Based Source Camera Statistical Certification". In: *2023 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE. 2023, pp. 1–6.

[GFF09] Miroslav Goljan, Jessica Fridrich, and Tomás Filler. "Large scale test of sensor fingerprint camera identification". In: vol. 7254. Feb. 2009, p. 72540. DOI: 10.1117/12.805701.

[Gol08] Miroslav Goljan. "Digital camera identification from images-Estimating false acceptance probability". In: *Proc. 8th Int. Workshop Digital Watermarking* (Jan. 2008), pp. 454–468.

[HJA20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[IFP21] Massimo Iuliani, Marco Fontani, and Alessandro Piva. "A Leak in PRNU Based Source Identification—Questioning Fingerprint Uniqueness". In: *IEEE Access* 9 (2021), pp. 52455–52463. DOI: 10.1109/access.2021.3070478. URL: https://doi.org/10.1109%2Faccess.2021.3070478.

[Jäh10] Bernd Jähne. "Emva 1288 standard for machine vision: Objective specification of vital camera data". In: *Optik & Photonik* 5.1 (2010), pp. 53–54.

[JFM20] Qiyu Jin, Gabriele Facciolo, and Jean-Michel Morel. "A review of an old dilemma: Demosaicking first, or denoising first?" In: *proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2020, pp. 514–515.

[JH12] Xiaodan Jin and Keigo Hirakawa. "Analysis and processing of pixel binning for color image sensor". In: *EURASIP Journal on Advances in Signal Processing* 2012 (2012), pp. 1–15.

[Lec+22] Bruno Lecouat et al. "High dynamic range and super-resolution from raw image bursts". In: *arXiv preprint arXiv:2207.14671* (2022).

[LFE23] Jamy Lafenetre, Gabriele Facciolo, and Thomas Eboli. "Implementing handheld burst super-resolution". In: (2023).

[LFG06] J. Lukas, J. Fridrich, and M. Goljan. "Digital camera identification from sensor pattern noise". In: *IEEE Trans. on Information Forensics and Security* 1.2 (2006), pp. 205–214. DOI: 10.1109/TIFS.2006.873602.

[Li+22] Haoying Li et al. "Srdiff: Single image super-resolution with diffusion probabilistic models". In: *Neurocomputing* 479 (2022), pp. 47–59.

[Loi+24a] Benjamin Loison et al. "Photo response non-uniformity extraction". In: 2024. URL: https://www.overleaf.com/read/msmjkfxtvptj#e2642a.

[Loi+24b] Benjamin Loison et al. *Photo response non-uniformity extraction demo*. July 2024. URL: https://ipolcore.ipol.im/demo/clientApp/demo.html?id=77777000503.

[Loi24a] Benjamin Loison. *Part of the source code of this paper*. July 2024. URL: https://gitea.lemnoslife.com/Benjamin_Loison/Robust_image_source_identification_on_modern_smartphones.

[Loi24b] Benjamin Loison. *Part of the source code of this paper*. July 2024. URL: https://codeberg.org/Benjamin_Loison/PRNU_extraction.

[Maı17] H. Maıtre. *From Photon to Pixel: The Digital Camera Handbook*. Digital signal and image processing series. Wiley, 2017. ISBN: 9781786301376. URL: https://books.google.fr/books?id=1-sCDgAAQBAJ.

[TM98] C. Tomasi and R. Manduchi. “Bilateral filtering for gray and color images”. In: *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*. 1998, pp. 839–846. DOI: 10.1109/ICCV.1998.710815.