https://doi.org/10.2352/ISSN.2470-1173.2021.4.MWSF-336

Camera Fingerprint estimation with a Generative Adversarial Network (GAN)

Sujoy Chakraborty; Department of Computer Science, Stockton University; Galloway, NJ/USA

Abstract

For forensic analysis of digital images or videos, the PRNU or camera fingerprint is the most important characteristics, for source attribution and manipulation localization. Typically, a good estimate of the PRNU is obtained by computing its Maximum Likelihood estimate from noise residuals of a large number of flat-field images captured by the camera. In this paper, we propose a novel approach of estimating the fingerprint of a camera, with a Generative Adversarial Network (GAN). The idea is to let the Generator network learn a distribution, from which PRNU samples will be drawn after training of the two adversarial networks. Experimental results indicate that the GAN-generated PRNU yields state-of-the-art camera identification and manipulation localization results.

Introduction

Digital camera sensor noise has been widely accepted by researchers from the forensics community as the most valuable characteristics for the forensic analysis of digital images [1, 2, 15]. Minute imperfections in silicon wafers and manufacturing inconsistencies give rise to a unique noise pattern, which is present in every image captured by a camera. This noise pattern, which is spatially varying, is termed as Photo-Response Non-Uniformity (PRNU) and can be estimated and tested for in forensic applications. It is also known as the "Fingerprint" of a camera, due to its uniqueness and due to the fact that it is left as a trace in every sensor output. For forensic analysis of digital images, the PRNU has been proven to be the most effective attribute, especially for the tasks of source attribution and manipulation localization.

The problem of source attribution is deeply investigated in the forensic research community, which entails tracing back the source of a digital image or video, i.e. to be able to identify the device that was used to capture a given image or video [5–15]. This is of particular importance in forensic scenarios, because it enables us to trace back the owner of a digital content, which helps us to fight cases such copyright infringement or distribution of illicit materials (such as under-age clips, terrorist threats etc.). The fundamental technique for source attribution is to inspect the query image for presence or absence of traces of the fingerprint of the reference camera. Typically, the detection statistic is the normalized correlation computed between the sensor noise extracted from the query image (also known as the noise residual) and the reference noise pattern (PRNU) of the digital camera device [15]. A different test statistic is the Peak-to-Correlation-Energy (PCE), which is often a more reliable statistic for camera identification [7, 11].

Another important forensic application is localizing manipulations in digital images. When the content of a part in a source image is replaced by content copied from a different image (splicing), or from another portion of the same image (copy-move), it replaces the sensor noise that was originally present in that region. A fundamental correlation-based detector inspects the query image in small overlapping analysis windows and compares the local noise with the corresponding part in the reference pattern, in terms of a correlation score. A correlation below some suitable threshold indicates a potential manipulation [15]. More advanced detectors exploit neighborhood dependency with random fields to improve localization performance [4, 19, 20].

For both the tasks of camera identification, as well as manipulation localization, the algorithms rely on an adequate estimate of the reference noise pattern. Often a reliable estimate of the sensor noise from the query image is a challenging task, especially if the query image has textured content [23], although advanced denoising algorithms have shown to mitigate this issue to some extent [4, 27, 28]. Also, the quality of the reference noise pattern might be questionable when we have a limited number of images available from the camera (for example, downloaded from social media), to compute the reference noise pattern. Hence, it is needless to mention that both the camera identification as well as manipulation localization algorithms would benefit from a high quality estimate of the sensor noise pattern, both from the query image, as well as for the reference pattern of the camera.

In this work, we focus on the second problem, that is of improving the estimate of the reference noise pattern of the camera. Typically, the PRNU of a camera is estimated from a set of genuine images captured by a given device, by extracting the noise residual from each image and then computing a weighted average. In this work, we propose a novel approach for obtaining the reference noise pattern of a camera, based on generative modeling. The idea here is to train a Generative Adversarial Network (GAN), so that the generator network learns a distribution from which new PRNU samples will be drawn, which can then be used for the tasks of camera identification and manipulation localization. Before we elaborate on the proposed approach, we provide a brief review of PRNU-based image forensics in the next section, along with a literature review on the relevant prior work that has been done to improve the sensor noise pattern in a forensic scenario. Next, we describe our proposed approach which includes the architecture of the two adversarial networks, tuning the hyperparameters and training the networks. Subsequently, we describe our experimental setup, results and conclusion.

Camera sensor-based Image Forensics

Let, $\mathbf{x} \in R^N$ denotes a genuine sensor output from a camera, and x_i denotes the intensity at the i-th pixel in the image in column-major order. Typically, a simplified model is assumed of the form:

$$\mathbf{x} = (1 + \mathrm{k}).\mathbf{x}^{(0)} + \boldsymbol{\theta} \tag{1}$$

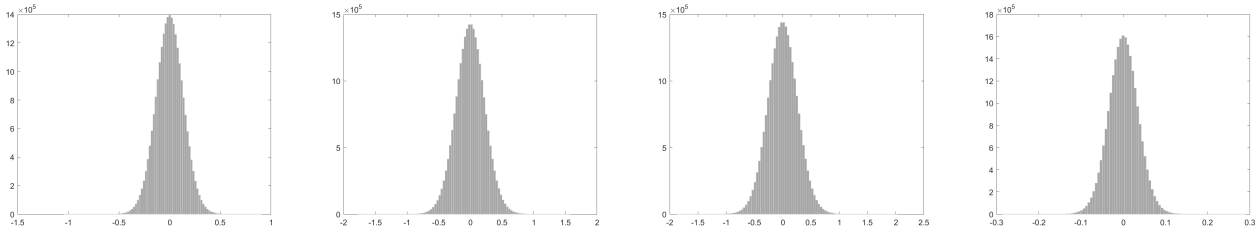


Figure 1: The distribution of $\varepsilon(m,n) = RP(m) - RP(n)$ for various values of m and n. From left to right: $m = 100, n = 50$; $m = 50, n = 20$; $m = 107, n = 20$; $m = 107, n = 100$.

where $\mathbf{x}^{(0)}$ represents the ideal sensor output, k denotes the camera fingerprint (PRNU) and θ represents the noise term that includes all various types of disturbances [3], which is typically modeled as i.i.d Gaussian. The PRNU factor k can be estimated from a set of genuine sensor outputs from the device, each of which is assumed to have a trace of the fingerprint factor k, specific to that device of interest. If $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L$ denote $\mathbf{L}$ genuine images captured by the camera, then the Maximum Likelihood Estimate (MLE) of the PRNU factor k can be computed [3] as:

$$\hat{\mathbf{k}} = \left(\sum_{l=1}^{L} \mathbf{w}_l \mathbf{x}_l \right) \cdot \left(\sum_{l=1}^{L} \mathbf{x}_l^2 \right)^{-1}, \tag{2}$$

where, $\mathbf{w}_l$ is the noise residual obtained by feeding the input image $\mathbf{x}_l$ to a suitable denoising filter $F(.)$: $\mathbf{w}_l = \mathbf{x}_l - F(\mathbf{x}_l)$. The weighting term $\mathbf{x}_l$ makes sure that the dark areas, in which PRNU is attenuated, contribute less to the overall estimate. A post-processing step is often applied to remove non-unique artifacts [3, 16] from the estimate.

The task of camera identification can be viewed as a hypothesis testing problem, where a query image is tested for the presence or absence of the fingerprint of the camera of interest:

$$\begin{cases} H_0 : \mathbf{w} = \mathbf{x} - F(\mathbf{x}) \text{ does not contain the PRNU } \mathbf{k} \\ H_1 : \mathbf{w} \text{ contains the PRNU factor } \mathbf{k} \end{cases}$$

where, H_0 is the null hypothesis which represents that the query image doesn't come from the camera under test. The hypothesis H_1 represents the alternative hypothesis, i.e., the query image indeed comes from the camera of interest. The hypothesis test can be decided in favor of H_0 or H_1 based on a computed correlation statistic and comparing the same to a predetermined threshold τ:

$$\rho = \text{corr}\left(\mathbf{w}, (\mathbf{x}\hat{\mathbf{k}})\right). \tag{3}$$

where, the standard algorithm decides for H_1 if $\rho > \tau$ and for H_0 otherwise. A more stable and reliable detection statistic, however, is the Peak to Correlation Energy measure (PCE) as reported in [7].

A manipulation of type copy-move or splicing, where a portion of the image is replaced by a different content, causes the original underlying fingerprint to be distorted or removed. Thus, such manipulations can be localized in an image by inspecting a query image in small overlapping analysis windows centering every pixel and computing a similarity score of the local noise estimate with the corresponding region in the reference fingerprint estimate $\hat{\mathbf{k}}$. Localization of small manipulations requires selection of a small analysis window. Typically, the literature recommends 64×64 analysis window as a reasonable trade-off between resolution and accuracy of localization [4, 19, 20]. Another problem that is reported in the literature is that the local correlation score is content-dependent and likely to be considerably low even in absence of manipulation, if the content under inspection is textured, dark or has saturated pixels. Chen proposes a remedy to this problem in terms of a correlation predictor $\hat{\rho}(\mathbf{x})$ [15], which predicts the expected correlation (assuming the content to be genuine) as a linear function of features representing the texture, intensity, saturation and flatness of the content:

$$\hat{\rho} = \sum_c \beta_c \cdot \phi_c(\mathbf{x}), \tag{4}$$

where, the linear regression coefficients β_c can be obtained from a set of genuine image patches with a least square fit [15]. Korus and Huang [19] used a feed forward neural network that was trained with the same features computed for a training set consisting of known images. In another recent work [21], a convolutional neural network (CNN) was adapted to automatically learn features, instead of using the hand-crafted features for predicting correlation.

Both the forensic tasks of camera identification as well as manipulation localization are benefitted from a high quality estimate of the reference fingerprint, as well as that of the residual noise from a query image. The recent success of data driven approach has motivated researchers to come up with deep neural network architectures, in particular, employing the Convolutional Neural Network (CNN) as a non-linear optimization tool, for the purpose of extracting noise from an image. Zhang et. al. [22] proposed a de-noising convolutional neural network (DnCNN), which suppresses the image and generates the noise at the output. Kirchner and Johnson propose the SPN-CNN [23], which learns to extract the sensor noise from a probe image under analysis. The SPN-CNN uses a pre-computed fingerprint $\hat{\mathbf{k}}$, and the network learns to generate the noise residual from the probe image by minimizing its distance from the target estimate $\hat{\mathbf{k}}$.

In this work, however, we focus on improving the reference noise pattern of the camera, or the estimate $\hat{\mathbf{k}}$. The idea here is to follow a generative modeling approach where a target distribution would be learnt by a deep neural network, and then will generate new samples from the learnt distribution. This deviates from the conventional weighted averaging of the noise residuals, which yields a stand-alone estimation of the PRNU, given a fixed set of flat-field images. In the next section, we describe our proposed approach in detail.

Estimating PRNU with GAN

The Generative Adversarial Network (GAN) is a recent exciting and promising generative modeling framework in Machine

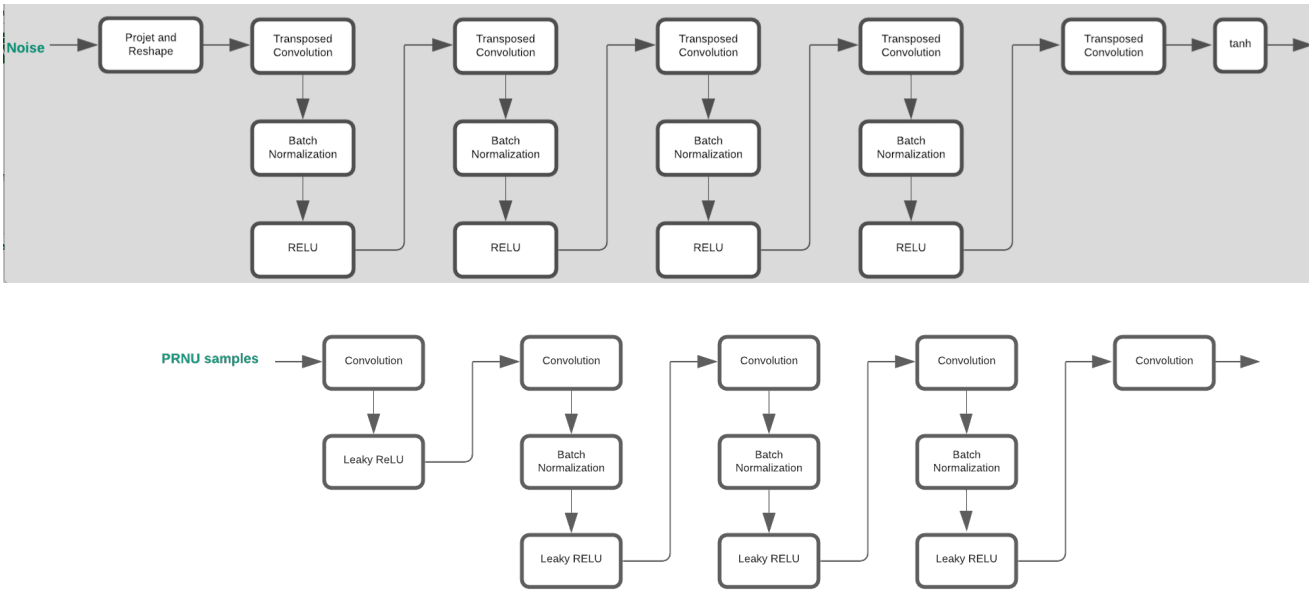


Figure 2: The architecture of the proposed Generator and the Discriminator networks.

Learning [24]. In a GAN, two deep neural networks compete with each other in an adversarial setup. One is the "Generator", which aims to learn a target distribution and generates samples from the distribution as it learns. The other is the "Discriminator", which aims to classify the samples generated by the generator as "fake". As the training of the two adversarial networks progresses, each network gets better and better in its own task. The generator gets better in generating samples that resembles the target distribution, whereas the discriminator gets better in classifying the generated samples as fake. After sufficient number of epochs, the generator becomes adequately trained and generates samples which the discriminator can no longer classify as fake, when the training is complete. Then the "Generator" network can be used to generate new samples from the target distribution.

For the fingerprint estimation, we propose a PRNU-GAN, with a goal to learn a target distribution P_{target}. After the generator is sufficiently well trained, it learns to "generate" PRNU samples from the distribution P_{target}, which can be used for camera identification, or manipulation localization. For the purpose of PRNU estimation, the distribution P_{target} is the distribution of PRNU samples, where each sample corresponds to the PRNU estimate obtained from a given number of genuine sensor outputs.

Let $RP(n)$ denotes the Maximum Likelihood estimate of the PRNU that we obtain from n images. Also, let $\varepsilon(m,n) = RP(m) - RP(n)$, which is the difference of the two PRNU samples, which are obtained respectively from m and n genuine images from the camera. We look at the distribution of $\varepsilon(m,n)$ for different values of m and n. Fig. 1 shows the distribution of $\varepsilon(m,n)$ respectively for 4 different pairs of (m,n). We observe that the error term $\varepsilon(m,n)$ follows a distribution, which can be very well approximated with a Gaussian distribution. We observe that the variance of the distribution of ε is larger when $|m-n|$ is larger. To generate the training data for the two adversarial networks, we started with an initial pre-computed estimate of the PRNU factor $\mathbf{k}$. We generate the training samples by adding to the pre-computed PRNU estimate a Gaussian random noise component which has a random variance for every generated sample:

$$k_t = k_I + \sigma.N, \tag{5}$$

where, k_I is the initial estimate of the PRNU factor $\mathbf{k}$, N is a Gaussian random noise component drawn from the standard normal distribution, i.e., $N \sim \mathcal{N}(0,1)$ and σ is a random variable drawn from a uniform distribution, which controls the variance of the added Gaussian noise component. Fig. 2 shows the architecture of the proposed generator and the discriminator networks. We considered to work with input sample size of 64×64 for faster training of the GAN. The input to the generator network is an array of random noise of size $[1 \times 1 \times 100]$. This is upscaled by four stages of transposed convolution, batch normalization and RELU non-linearity, followed by a final transposed convolution layer. The output of the final transposed convolution layer is of dimension $64 \times 64 \times 3$, where we generate the PRNU samples for all three color channels. The output layer uses a tanh activation function. In the generator network, for each transposed convolution layer, we used filters of size 4×4 and a stride of 2, which upsamples the input by a factor of 2. In the first 4 transposed convolution layers, we respectively use 512, 256, 128 and 64 filters. The final transposed convolution layer has 3 filters, which correspond to the three color channels. Xavier initialization has been used to initialize the weights in the transposed convolution layers. The discriminator takes the generated PRNU samples and compares it to the training set of samples. The input to the discriminator are $64 \times 64 \times 3$ arrays, which then pass through a series of convolution layers combined with batch normalization and leaky RELU non-linearity. The output of the discriminator is a scalar prediction score, which is the confidence score or probability with which the network classifies that generated sample as "fake". For each convolution layer in the discriminator network, we used filters of size 4×4. The number of filters in the first 4 convolution layers are respectively 64, 128, 256 and 512 respectively. We use again

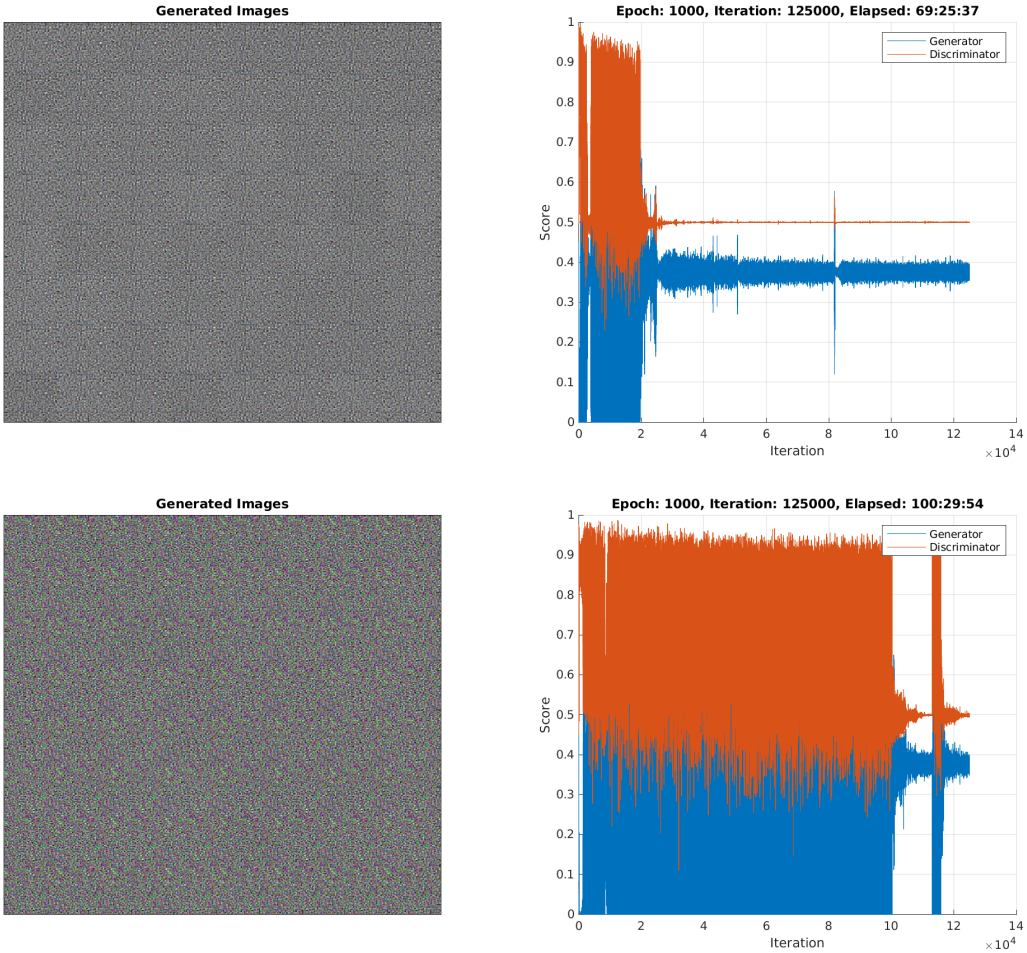


Figure 3: Training of the Generator and the Discriminator networks. Top: Canon 60D (Realistic Tampering dataset) and Bottom: Olympus Mju 1050SW (Dresden dataset)

the Xavier initialization to initialize weights in the convolution layers.

Experiments

We worked with a total of 22 cameras from two different datasets for our experiments, the Dresden dataset [17] and the Realistic Tempering dataset [19], to illustrate the efficacy of the GAN-generated PRNU samples for camera identification and manipulation localization.

Training

As mentioned earlier, we found that the network takes a very long time for training, when the input patch size is large. Even with a patch size of 256×256, we could only reach close to 100 epochs after 5 days of training. The training time is dependent on the hardware as well, but as a general rule of thumb, the training time increases rapidly, as we increase the dimension of the input sample size. This forced us to restrict our input sample size to 64×64. We divided the PRNU of a camera into multiple parts, each of dimension 64×64 and trained for each part individually. This reduced the training time considerably. To generate the training data, following equation (5), we generated 1000 training samples corresponding to each part of size 64×64 of the PRNU, for each camera that we worked with. The initial estimate of the fingerprint k_I was computed with a set of genuine images from each camera. We used a learning rate of 0.0001 with a gradient decay factor of 0.5 for both the generator and the discriminator networks. We found the GAN to be extremely sensitive to the choice of hyperparameters, which often leads to convergence issues.

Fig. 3 shows the training of two different cameras: a Canon 60D from the Realistic Tampering dataset [19] and an Olympus Mju 1050SW, from the Dresden dataset [17]. We trained both the networks for 1000 epochs. For all cases, we use a batch size of 8. We observe that after 1000 epochs, the generator is so well trained that it generates samples which the discriminator fails to classify as fake any more. The output score from the discriminator is 0.5, which corresponds to random guessing. However, one point that is important to mention here is that, it is not the case that every time we attain an equilibrium after 1000 epochs. It can be the case (which actually is the case for most of the times) that even after 1000 epochs, the discriminator is able to classify the generated samples as ”fake”, but even then, we found that the generator is sufficiently well trained after 1000 epochs that it generates PRNU samples of adequate quality, for camera identification and

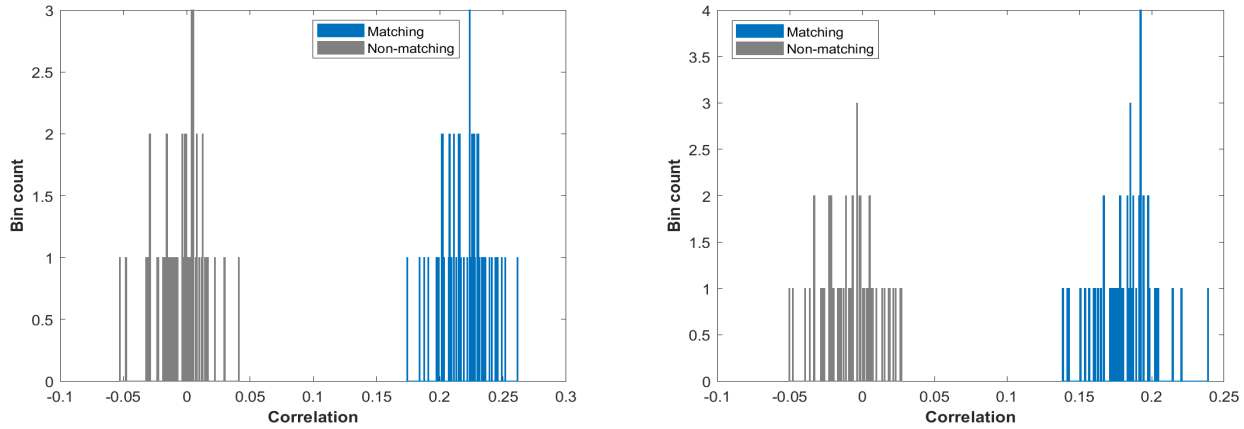


Figure 4: Correlation of noise residuals with GAN-generated fingerprint and fingerprint of a non-matching camera. Left: Canon IXUS 70 and Right: Olympus Mju 1050SW.

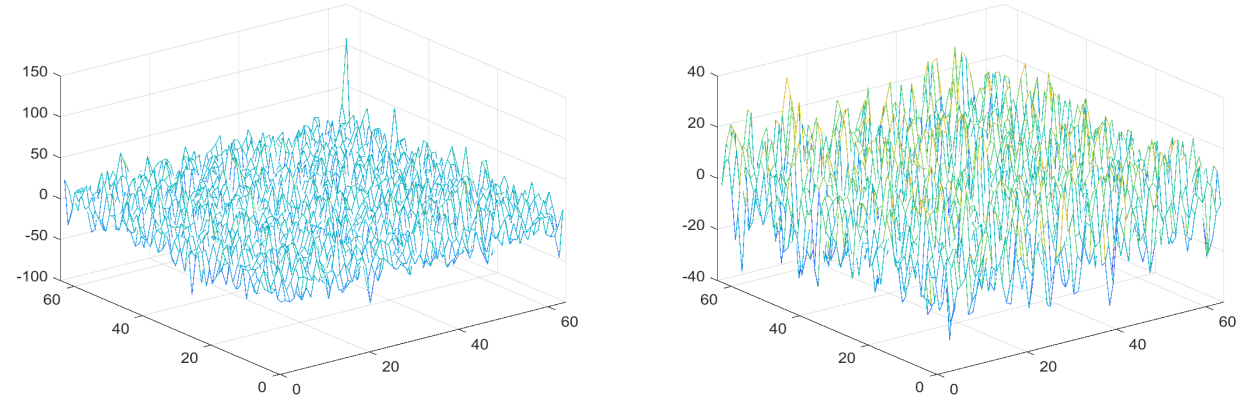


Figure 5: Comparison of PCE peak for a sample noise residual (64×64) from a Nikon D7000 camera, obtained with Left: GAN-generated PRNU and Right: Initial PRNU estimate

manipulation localization.

Results

To demonstrate the efficacy of the GAN-generated fingerprints, we show here baseline results for the accuracy of camera identification. Fig. 4 shows results for two different camera devices, but similar results can be obtained for other devices as well. To the left, we show the distribution of correlations computed for noise residuals of 50 images from a Canon IXUS 70 camera (Dresden dataset), with a PRNU sample generated by the proposed Generative Adversarial Network for the same camera, along with the correlations computed with a non-matching PRNU (that of a Canon 60D, Realistic Tampering dataset). To the right, we show the correlations of noise residuals of images from an Olympus MJU 1050 SW, with a GAN generated PRNU sample and correlations computed with a non-matching camera PRNU (Pentax Optio A40). Both cameras are from the Dresden dataset in this case. The patch size is 64×64, which corresponds to one specific part of the PRNU and the residuals. We observe that the two distributions are very distinctly separable and camera identification works perfectly for both the cases. Fig. 5 shows the PCE peak comparison for a patch of a noise residual with a GAN-generated PRNU sample (left) and a stand-alone PRNU sample computed by weighted averaging, for a sample query image from a Nikon D7000 camera. The patch size in this case is again 64×64. It is evident that the GAN-PRNU yields a much higher peak in this case, which is primarily due to the reason of a better estimate of the reference noise pattern.

In Fig. 6, we show the comparison of the PCE values for 19 sample images from an Olympus MJU 1050 SW camera from the Dresden dataset, as well as the comparison of PCE peak height for 50 images from a Praktica DCZ5.9 camera. For each plot, we show the comparison of the computed statistic for the GAN-generated PRNU pattern and the pattern computed with weighted averaging over a fixed set of noise residuals. We observe that the GAN-generated PRNU samples could yield higher PCE values as well as higher PCE peak heights for both the devices. Similar results can be obtained with other devices from the two datasets.

Fig. 7 shows the PCE values obtained with 50 images each from a FujiFilm FinePixJ50 and an Olympus mju 1050SW camera respectively, for a patch of size 64×64. We show PCE values obtained with the matching and 4 other non-matching PRNU-s, where all PRNU samples are GAN-generated. We observe that the GAN-generated PRNU-s yield clearly separable distributions for matching and non-matching cases. We obtain similar results for other cameras in our test database.

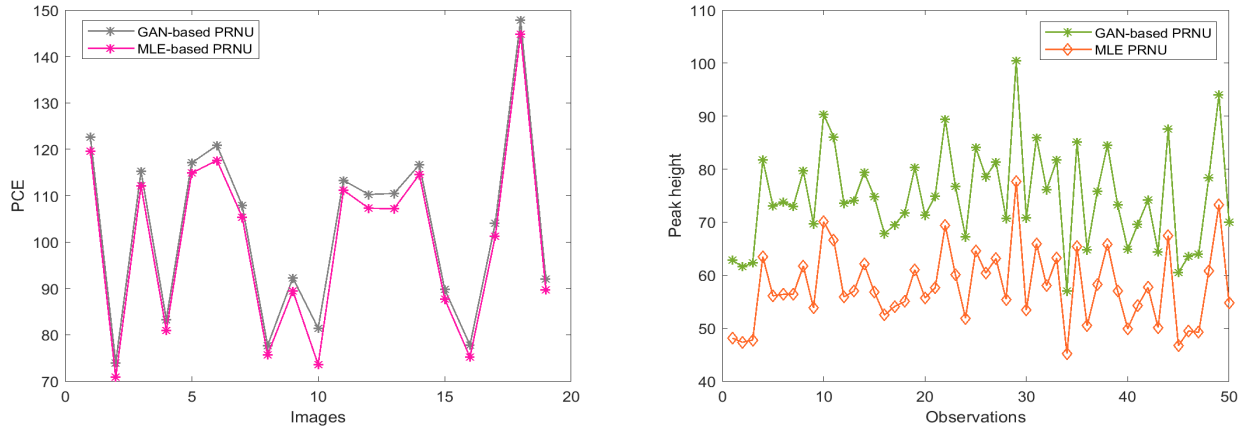


Figure 6: Comparison of PCE attributes for GAN-based PRNU and an initial PRNU estimate obtained by weighted averageing . Left: PCE values (Olympus MJU 1050 SW) and Right: PCE peak height (Praktica DCZ5.9)

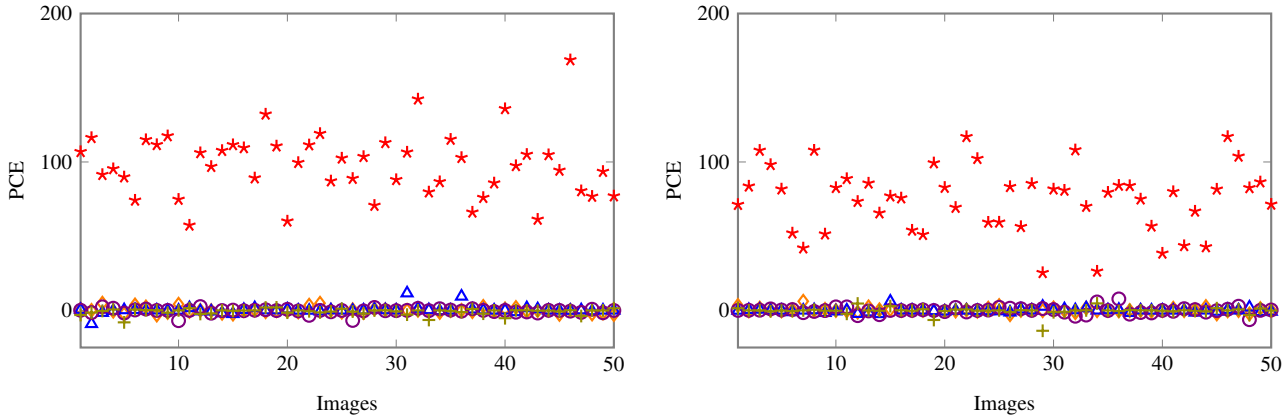


Figure 7: PCE values for patch size of 64×64 for two different cameras for matching and non-matching cases. Left: Olympus mju 1050SW and Right: FujiFilm FinePixJ50. Cameras: ⋆Matching camera, ◇Ricoh GX100, △Nikon D200, ○Panasonic DMC FZ50, ◇Pentax OptioA40

In Fig. 8, we show the PCE values of matching and non-matching PRNU-s for patch size of 512×512, for a Canon 60D (left) and a Nikon Coolpix S710 camera (right). The PRNU of both the cameras were obtained by combining a GAN-generated sample for each part of the PRNU. For the Canon 60D, we show the PCE values with the noise residuals from the three other devices (Nikon D90, Nikon D7000 and Sony A57), all from the Realistic Tampering dataset. For the Nikon Coolpix S710, the non-matching cameras are Olympus mju 1050SW, Pentax Optio A40 and Samsung L74 Wide, all 4 cameras are from the Dresden dataset. We observe that the PCE values are distinctly separable, for matching and non-matching cases and camera identification works perfectly. Also, for the matching cases, the PCE values for 512×512 patches are much higher than the PCE values for a patch size of 64×64, as expected.

Fig. 9 shows the baseline ROC for manipulation localization for some images from a Canon 60D camera, where the correlation has been computed with the GAN-generated PRNU sample, for a 512×512 patch size. We also used correlations computed with a stand-alone PRNU sample, obtained with weighted averaging over noise residuals, for comparison. The results indicate that GAN-based fingerprints yield state-of-the-art baseline results, which could be further enhanced with recent sophisticated random field based detectors [4, 19, 20].

Conclusion

We demonstrate that it is possible to improve the estimate of the reference noise pattern by letting a generative adversarial network learn a distribution of the PRNU samples. We demonstrate the efficacy of the proposed approach with large scale experiments conducted on 22 cameras from the two popular datasets used for forensic research: the Dresden dataset and the Realistic Tempering dataset. We observe that GAN-generated fingerprints improved upon the initial estimate with which we started. However, further research is needed to address the challenges we faced during training. One issue with GAN is the training time which is considerably large for larger patch size, which forced us to divide the PRNU in multiple non-overlapping parts and train for each part individually. Future research will target to mitigate this challenge. We also found that the GAN is extremely sensitive to even slightest variation in hyperparameters, for instance, we found convergence issues even with a learning rate as low as 2×10^{-4}, which motivates us to further investigate the effect of fine tuning the hyperparameters on the network performance.

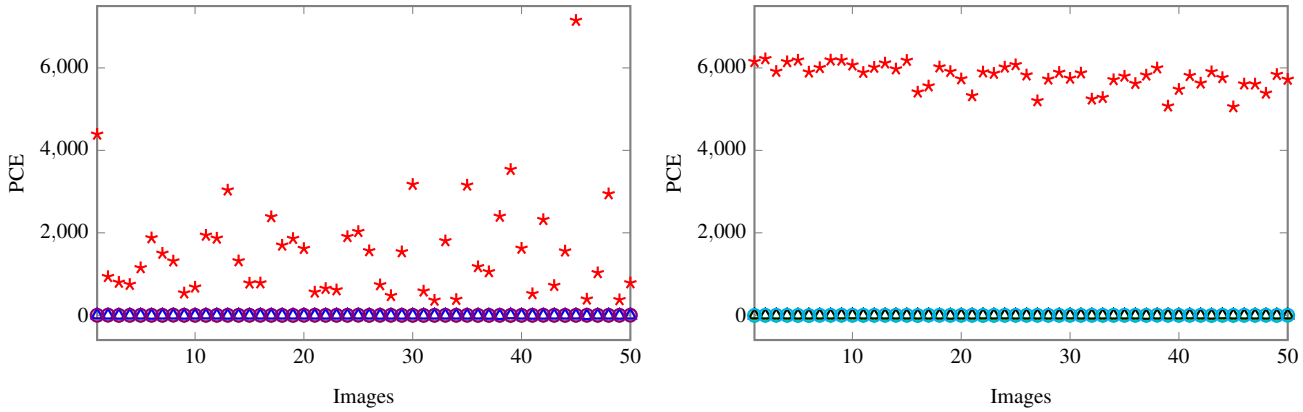


Figure 8: PCE values for patch size of 512×512 for two different cameras for matching and non-matching cases. Left: Canon 60D and Right: Nikon Coolpix S710. Cameras: ⋆Matching camera, ◇Nikon D90, △Nikon D7000, ○Sony A57, ◇Olympus mju 1050SW, △Pentax OptioA40, ○Samsung L74 wide

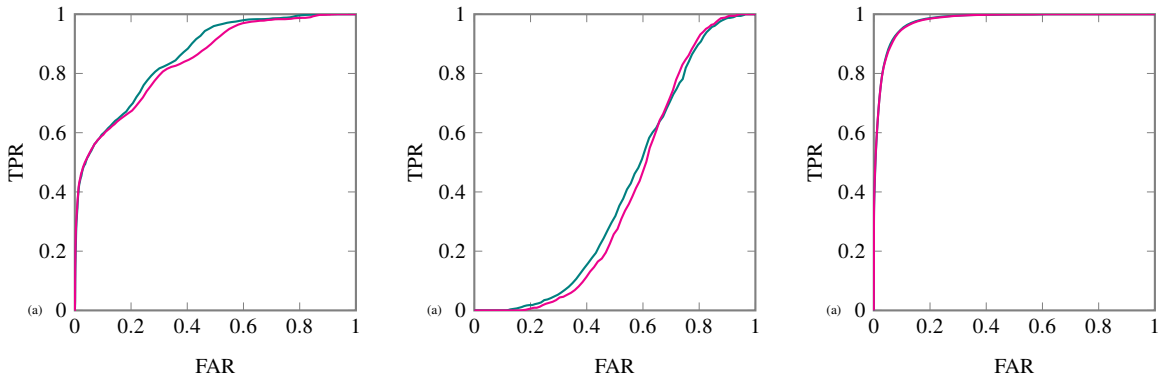


Figure 9: Image manipulation localization ROC curves for some images from a Canon 60D camera: GAN-based——, Initial MLE-based ——.

References

[1] R. Böhme and M. Kirchner, "Media forensics," in *Information Hiding*, S. Katzenbeisser and F. Petitcolas, Eds. Artech House, 2016, ch. 9, pp. 231–259.

[2] J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.

[3] J. Fridrich, "Sensor defects in digital image forensics," in *Digital Image Forensics: There is More to a Picture Than Meets the Eye*, H. T. Sencar and N. Memon, Eds. Springer, 2013, pp. 179–218.

[4] G. Chierchia, G. Poggi, C. Sansone, and L. Verdoliva, "A Bayesian-MRF approach for PRNU-based image forgery detection," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 554–567, 2014.

[5] M. Chen, J. Fridrich, and M. Goljan, "Source digital camcorder identification using ccd photo response nonuniformity," in Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Mul- timedia Contents IX, E. Delp and P. Wong, Eds., vol. 6505, January 2007, pp. 1G 1–12.

[6] T. Filler, J. Fridrich, and M. Goljan, "Using sensor pattern noise for camera model identification," in Proc. ICIP 2008, San Diego, CA, October 12-15 2008, pp. 1296–1299.

[7] M. Goljan , "Digital camera identification from images—Estimating false acceptance probability," in Proc. 7th Int. Workshop Digital Watermarking, Busan, Korea, Nov. 10–12, 2008.

[8] M. Goljan and J. Fridrich, "Camera identification from scaled and cropped images," in Proceedings of SPIE, Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X, E. J. D. et al., Ed., vol. 6819. San Francisco, CA: SPIE, January 2008, pp. 0E 1–13.

[9] M. Goljan and J. Fridrich,"Sensor-fingerprint based identification of images corrected for lens distortion," in Proceedings of SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2012, N. Memon, A. Alattar, and E. D. III, Eds., vol. 8303, San Francisco, CA, January 2012, pp. 0H 1–13.

[10] M. Goljan, J. Fridrich, and M. Chen, "Sensor noise camera identification: Countering counter-forensics," in Proc. SPIE Electronic Imaging, Steganography, Security, and Watermarking of Multimedia Contents XII, N. Memon, J. Dittmann, A. Alattar, and E. Delp, Eds., vol. 7541, January 17- 21, 2010, pp. 0S 1–12.

[11] M. Goljan, J. Fridrich, and T. Filler, "Large scale test of sensor fingerprint camera identification," in Proc. SPIE, San Jose, CA, Jan. 18–22, 2009, vol. 7254, pp. 0I 1–0I 12, Electronic Imaging, Media Forensics and Security XI.

[12] W. van Houten and Z. Geradts, "Source video camera identification for multiply compressed videos originating from youtube," Digital Investigation, vol. 6, no. 1-2, pp. 48–60, 2009.

[13] Y. Hu, C. Jian, and C.-T. Li, "Using improved imaging sensor pattern noise for source camera identification," in Multimedia and Expo (ICME), 2010 IEEE International Conference on, July 2010, pp. 1481–1486.

[14] D.-K. Hyun, S.-J. Ryu, M.-J. Lee, J.-H. Lee, H.-Y. Lee, and H.-

K. Lee, "Source camcorder identification from cropped and scaled videos," in Proc. SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics XIV, A. Alattar, N. D. Memon, and E. J. Delp, Eds., vol. 8303, January 23–26, 2012, pp. 0E 1–8.

[15] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, 2008.

[16] T. Gloe, S. Pfennig, and M. Kirchner, "Unexpected artefacts in PRNU-based camera identification: A 'Dresden Image Database' case-study," in *ACM Multimedia and Security Workshop (MM&Sec)*, 2012, pp. 109–114.

[17] T. Gloe and R. Böhme, "The Dresden Image Database for benchmarking digital image forensics," *Journal of Digital Forensic Practice*, vol. 3, no. 2–4, pp. 150–159, 2010.

[18] M. K. Mıhçak, I. Kozintsev, K. Ramchandran, and P. Moulin, "Low-complexity image denoising based on statistical modeling of Wavelet coefficients," *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 300–303, 1999.

[19] P. Korus and J. Huang, "Multi-scale analysis strategies in PRNU-based tampering localization," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 809–824,.

[20] S. Chakraborty and M. Kirchner. PRNU-based image manipulation localization with discriminative random fields. Electronic Imaging, 2017(7):113–120, 2017.

[21] S. Chakraborty. A CNN-Based Correlation Predictor for PRNU-Based Image Manipulation Localization. Electronic Imaging, 2020(7):pp. 78-1-78-8(8), 2020.

[22] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising," IEEE Transactions on Image Processing, vol. 26, no. 7, pp. 3142–3155, 2017.

[23] M. Kirchner and C. Johnson, "SPN-CNN: Boosting sensor-based source camera attribution with deep learning," presented at the IEEE Int. Workshop Inform. Forensics Secur., Delft, Netherlands, Dec. 9–12, 2019.

[24] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In Proceedings of NIPS, pp. 2672– 2680, 2014.

[25] D. Cozzolino, G. Poggi, and L. Verdoliva, "Splicebuster: A new blind image splicing detector," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2015.

[26] W. Fan, K. Wang, and F. Cayre, "General-purpose image forensics using patch likelihood under image statistical models," in *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2015.

[27] I. Amerini, R. Caldelli, V. Cappellini, F. Picchioni, and A. Piva, "Analysis of denoising filters for photo response non uniformity noise extraction in source camera identification," in International Conference on Digital Signal Processing (DSP), 2009, pp. 511–517.

[28] A. Cortiana, V. Conotter, G. Boato, and F. G. B. De Natale, "Performance comparison of denoising filters for source camera identification," in Media Watermarking, Security, and Forensics III, ser. Proceedings of SPIE, N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., vol. 7880, 2011.

Author Biography

Sujoy Chakraborty received his ME in Software Engineering from Jadavpur University, Kolkata (2008), India and his PhD in Electrical and Computer Engineering from the State University of New York, Binghamton (2019). He is currently working as an Assistant Professor of Computer Science at Stockton University, NJ. His work has focused primarily on digital image forensic techniques based on digital camera sensor noise, application of deep learning on sensor-based image forensics, as well as algorithms based on compression forensics and counter forensics.

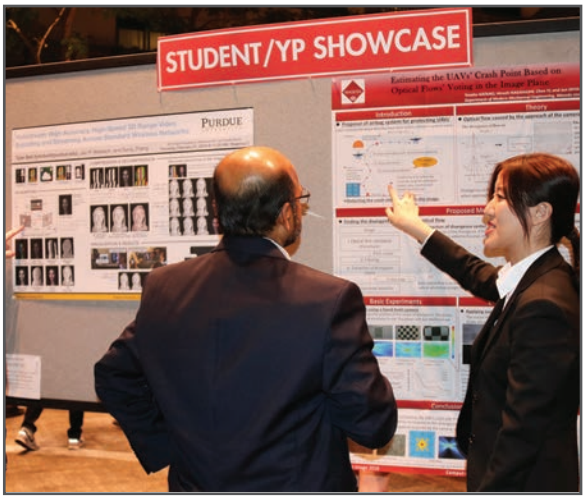
JOIN US AT THE NEXT EI!

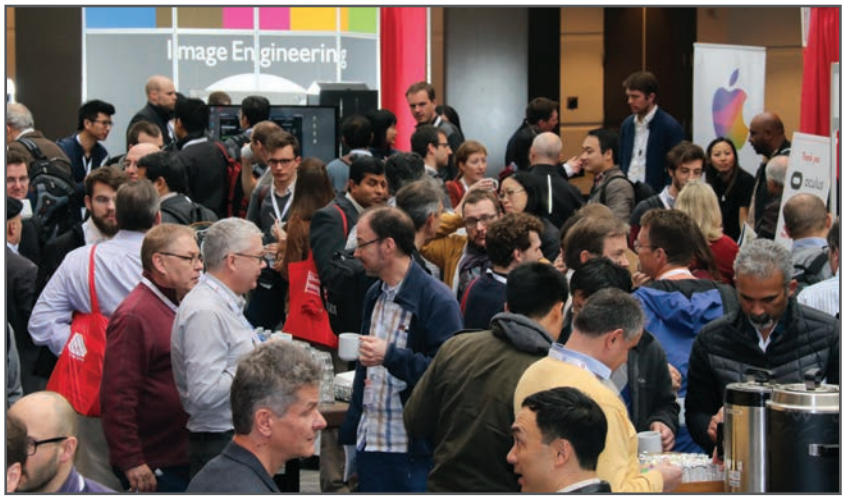
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org