A Study of the Robustness of PRNU-based Camera Identification

Kurt Rosenfeld and Husrev Taha Sencar
Polytechnic Institute of NYU, Brooklyn, NY

ABSTRACT

We investigate the robustness of PRNU-based camera identification in cases where the test images have been passed through common image processing operations. We address the issue of whether current camera identification systems remain effective in the presence of a non-technical, mildly evasive photographer who makes efforts at circumvention using only standard and/or freely available software. We study denoising, recompression, out-of-camera demosaicing.

1. MOTIVATION

Camera identification has emerged as a powerful tool for law enforcement. The objective of a camera identification system is, given a photograph, to identify the camera that took the photograph. This can be class identification (manufacturer and model), or it can be identification of an individual camera. This capability is of use in law enforcement when the case involves illegal images. The possessor of the camera that took the images could be considered a suspect.

Images pass through a variety of processing operations at various stages of their use. Rarely are they distributed without some or all of the following operations being applied: cropping, scaling, rotation, contrast enhancement, gamma correction, white balance correction, denoising, compression, and recompression. Additionally, common photo editing software invites more sophisticated manipulations like copy/paste, demosaicing, and image compositing. Most importantly, a photographer might deliberately take actions to hinder camera identification.

The purpose of this study is to evaluate the threat of identification circumvention. Our threat model is that the photographer has access to common commercially available photo editing software and will attempt to remove identifiable features from the image without significantly degrading subjective quality. By “attack” we mean distributing images without attribution. We are explicitly excluding the case of a determined attacker with a high level of resources and technical prowess since it accepted that such an attacker will succeed. Our goal is to determine what image processing operations are most problematic, why they are problematic, and whether the weakness can be eliminated. The robustness of PRNU-based camera identification has been discussed in previous work. Previous work assumed a non-hostile threat model. Our threat model differs in that it assumes that the photographer will make a deliberate effort to evade camera identification. Since our threat model does not assume that the photographer is an expert in signal processing, we believe that we are examining the most likely use-cases of camera identification in forensic work.

In the first section, we provide a brief overview of typical camera identification schemes. In the second section, we examine the effect of denoising on camera identification accuracy. In the third section, we examine the effect of recompression on camera identification accuracy. In the fourth section, we examine the effect of demosaicing on camera identification accuracy.

2. TYPICAL CAMERA IDENTIFICATION SCHEMES

Camera identification has been studied for decades. The objective of camera identification is to associate images with the camera that produced them. In recent years, this has been studied in the context of digital photography. A variety of features have been employed, including color filter array design, pixel color value interpolation algorithm, image sensor anomalies, lens characteristics and anomalies, and image processing pipeline characteristics. These techniques can be divided into two categories based on whether they provide identification of the just the camera model, or provide identification of the individual camera. For example, the algorithm that is
used for pixel color value interpolation does not vary across individual instances of a model of camera. Physical anomaly-based features like image sensor imperfections vary from one individual camera to the next, within one model. Clearly this second category of identification techniques is more powerful. Nevertheless, in practice, the various techniques each provide useful, albeit partial, information.

Currently, the dominant camera identification technique is based on photo response non-uniformity (PRNU). This feature is obtained by measuring the effects of semiconductor device-level anomalies in the image sensor of digital cameras. PRNU is a distinctive feature of each individual camera. It is not practical for the image processing pipeline in the camera to completely compensate for the anomalies in the sensor’s photo-response, so the artifacts are present in the digital images that it produces. Very high quality lossy compression (such as JPEG level 90) does not destroy the effectiveness of PRNU as a feature for camera identification. This is essential since, in practice, most digital cameras output their image as a high-quality JPEG.

The basic procedure for PRNU-based camera identification is composed of two phases; training and identification. Training is performed by building a fingerprint for each camera in the study. A fingerprint for a camera is constructed by obtaining several hundred photographs taken with the camera, extracting the noise from each image, and calculating a weighted average of these images. The noise is extracted by running a denoising algorithm on the image and subtracting the denoised image from the original image. Identification determines which, if any, camera in the study is likely to have produced the image in question. Identification begins by extracting the noise pattern estimate from the image in question. A comparison function is then invoked to determine the similarity of that image's noise pattern estimate to the each of the cameras' fingerprints. Early camera identification systems used normalized cross correlation to evaluate similarity.

\[
C(k) = \frac{\sum_{i=1}^{n} a_i b_{i+k}}{\sqrt{\sum_{i=1}^{n} a_i^2 \sum_{i=1}^{n} b_i^2}}
\]  
(1)

More recently, peak to correlation energy ratio (PCE) has been applied as the comparison function.

\[
PCE = \frac{c^2(0)}{\sum_{k \in A} \sum_{k \in A} c^2(k)}
\]  
(2)

where \(c\) is the cross-correlation of the image’s PRNU estimate with the camera’s PRNU fingerprint, and \(A\) is a small region around the origin that is excluded from the summation. PCE was first used in the context of recovery from cropping and scaling.\(^2\) More recently, Goljan makes a compelling argument for using PCE as the primary comparison function in camera identification systems.\(^3\) PCE has the benefit of being more consistent, reducing the need for frequent tuning of the detection threshold.

3. DENOISING

Denoising is frequently performed on images to improve their subjective quality. The objective of denoising is to increase subjective smoothness without attenuating high-spatial-frequency information in the image, such as edges and fine texture. Denoisers implicitly or explicitly assume a model for the noiseless image and a model for the noise, and attempt to separate these two components. At first glance, it might seem as though denoising would destroy the effectiveness of PRNU-based camera identification, since identification is done based on the noise. However, this is not the case. There are three reasons for this. First, denoising is not an absolute operation. All practical denoisers have tunable parameters that are set to optimize some sort of tradeoff. Even when the parameters are set to optimal values, some noise remains after denoising. The second reason is that there are many different denoising algorithms in use, and, given the same input, each one generates a distinct output. Each denoiser therefore removes different components from the image. Third, and most interesting, is that repeated denoising removes similar components at each iteration.
Figure 1. Multiple passes of wavelet-domain denoising are performed on a set of images. After each denoising, we observe the average normalized cross correlation between the image PRNU estimates and the camera PRNU fingerprint.

3.1 Denoising Experiments

To measure the effects of denoising, we investigate the effect of multiple passes of denoising on the correlation between the extracted PRNU pattern and the fingerprint for a camera. In the first phase of experiments, we used two different denoisers, the first one representing the processing done by the photographer, and the second one being the denoiser used in PRNU extraction in the camera identification system. This experimental setup is intended to give results that relate to a threat model of a photographer who wishes to circumvent camera identification but does not have access to (or knowledge of) the denoising algorithm used in camera identification.

3.2 Differing Denoisers

In our preliminary experiments, the attacker’s denoiser was a C-language reimplementation of a wavelet-domain denoiser by Selesnick. The denoiser was invoked repeatedly on a set of 300 images. At each stage, the PRNU pattern was extracted from each image, correlated with the PRNU fingerprint for that camera, and these 300 correlation values were averaged.

3.3 Identical Denoiser

Adopting a slightly different threat model, we assume that the attacker has access to a denoiser that is identical to that which is used for PRNU estimation in camera identification. In this case we investigated the extent to which processing with the identical denoiser is detrimental to camera identification. The relevance of this particular experiment is that it measures the value of secrecy regarding the camera identification system’s denoiser.

Our experimental observations were surprising. It makes little difference whether the same denoiser of a different denoiser is invoked by the attacker on the images. Intuitively, one might expect the denoising with the identical denoiser to remove the components from which PRNU is calculated. This is not the case. Indeed the residuals produced at successive iterations of denoising remain remarkably correlated with the residual produced at the first denoising. From this we conclude that the knowledge of the denoiser used for camera identification does not give the attacker any significant advantage for PRNU circumvention.
Figure 2. An image is denoised multiple times. At each iteration, a noise residual is produced. We observe the correlation if the n-th noise residual with the first noise residual. The figure shows the average behavior across 300 images. Note that the correlation does not fall fast.

4. RECOMPRESSION

Practically all digital cameras produce output in JPEG format by default. Many provide the option of producing output in so-called “raw” format. In practice, nearly all users leave their camera in the default mode. JPEG compression is performed by the camera as the final step of its image processing pipeline. Nearly all cameras give the user a set of options allowing them to choose their preferred point in the image quality versus file size tradeoff. The two main options are resolution and JPEG quality level. Most cameras default to maximum resolution and a medium-to-high JPEG quality level.

Recomposition is the image processing operation of decompressing an image, possibly modifying the uncompressed image, and recompressing it again. It is a known property of JPEG that recompression using the same quantization tables does not introduce further degradation. However, recompression with different quantization tables does introduce further degradation. This is the case even if the recompression is done using a higher JPEG quality level, (i.e. smaller values in the quantization tables). The reason for this is that the error introduced each time the image is compressed corresponds to the modulus of each DCT coefficient and its corresponding quantization table entry. After decompression, all DCT values are integer multiples of their corresponding quantization table entries. Recompression using the same quantization tables introduces no further degradation because the moduli of the DCT coefficients will already be zero.

Typically, when images are processed in photo editing programs, even if the input and output are both JPEG format, the quantization tables will differ. Consequently, error is introduced. Furthermore, if global image operations like brightening, contrast enhancement, or white balance correction are performed, error will be introduced by saving the data to JPEG, due to the fact that the moduli of the DCT coefficients will be nonzero. It is therefore to be expected that each time an image is processed in photo editing software, further degradation will take place.

4.1 Recompression Experiments

In our recompression experiments, we assumed that the goal of circumvention is to make accurate camera identification impossible while diminishing the quality of the image as little as possible. As an indication of the
effect on accurate identification, we observe the correlation of the PRNU pattern extracted from the test image with the camera’s fingerprint (PRNU) pattern. As an indication of the quality of the image, we use PSNR. We studied DCT- and wavelet-based compression and saw similar results. PRNU correlation is not eliminated by levels of compression that result in tolerable image quality.

5. DEMOSAICING

Another image processing operation that is available in common photo editing software is demosaicing. This operation is primarily intended as a higher-performance alternative to the built-in demosaicing performed by the camera. When digital photographs are taken in raw format, out-of-camera demosaicing is almost invariably performed. It can also be performed on images that were not captured in raw format, by dropping the interpolated color components of each pixel. In out-of-camera demosaicing, the pixel color component interpolation is performed by software on the computer. This gives the user more control over such details as the selection of the interpolation algorithm and the parameters of the algorithm.

The effects of out-of-camera demosaicing are relevant for three reasons. First, as a filtering operation, it can impair observation of PRNU. Second, since the demosaicing algorithm is a stable and distinctive class-level characteristic of cameras, disabling demosaicing in the camera and replacing it with post-processing in the computer takes away a useful feature that aids in camera identification. Third, a sufficiently aggressive demosaicing algorithm can, in fact, mask artifacts of the color filter array pattern in the camera, thereby impairing another class-level feature for camera identification. Since PRNU is currently the preferred feature for identification of individual cameras, we examine the effect of out-of-camera demosaicing on PRNU.

Our results show that a mismatch in the choice of demosaicing algorithm utilized in generation of images used during camera fingerprint extraction and matching might degrade detection accuracy. To compensate for this, one may need to extract a camera fingerprint for each demosaicing algorithm. However, the attacker may still circumvent this by demosaicing an image using a different demosaicing algorithm in different parts of a raw image.
bilinear VNG PPG AHD
set1 1.0000 0.7795 0.6837 0.7558
set2 0.1204 0.1094 0.0193 0.0739
set3 0.1259 0.1159 0.0157 0.0761
set4 0.1199 0.1076 0.0039 0.0671

Figure 4. A collection of 300 raw images was split into four sets. All images in each set were demosaiced with each of four different interpolation algorithms. The PRNU pattern was extracted for each set/algorithm pair. We observe the correlation of the set1/bilinear PRNU with the other PRNUs. With raw files, the photographer can choose the interpolation algorithm, which could impair camera identification.

6. LIMITATIONS

The Stirmark package is typically used for evaluating the robustness of watermarking algorithms. The threat model there is that some media, typically a photo, has a watermark embedded in it, and an attacker wants to remove the watermark and redistribute it. The attacker’s main criteria for watermark removal are that it should adequately reduce the probability of correct watermark detection in the modified image, and the perceptual quality of the modified image should not be significantly degraded. Stirmark is essentially a programmable set of subtle image distortions: deleting rows, rotating, etc. Naive watermarking algorithms are completely circumvented by Stirmark. Robust watermarking algorithms can tolerate a greater amount of Stirmark’s distortions while retaining high detection probability. Since the attack model in camera identification circumvention closely resembles the attack model in image watermark circumvention, we examined the effects of the Stirmark operations on a typical PRNU-based camera identification system.

In its standard (naive) state, PRNU-based camera identification is extremely fragile. If it is thought that cropping and/or scaling might have been performed on the images, a recovery/resynchronization search can be performed. Otherwise, a false non-match will result from these operations. High complexity operations like the random bending attack (RBA) in Stirmark have no known recovery/reversal algorithm. Similar effects are available in software like Photoshop. A non-evasive photographer is unlikely to invoke one of these effects, but a mildly evasive photographer could make use of them for PRNU circumvention. Studying RBA-like operations would require either human subjects or a sophisticated model of human visual perception. PSNR will give a pessimistic estimate of the subjective degradation, due to the local misalignments caused by the bending.

In general, an attack that works against watermarks will work at least as well against PRNU. The reason for this is rather straightforward. The intensity values of the pixels in a PRNU pattern can be considered to be IID. Therefore the 2-D autocorrelation of the PRNU pattern is a 2-D delta function. Shifts of and image caused by Stirmark result in corresponding shifts in the PRNU estimate for that same image. Camera identification is based on the projection of the PRNU estimate for an image on the PRNU fingerprint for a camera. The projection of a shifted delta function onto an unshifted delta function is zero. Experimental results confirm this.

REFERENCES
