A Study of the Robustness of PRNU-based Camera Identification

Kurt Rosenfeld Taha Sencar Nasir Memon

Abstract

We investigate the robustness of PRNU-based camera identification in cases where the test images have been passed through common image processing operations. A specific question we aim to answer is how camera identification can be circumvented by an nontechnical user, applying only standard and/or freely available software. We study denoising, recompression, out-of-camera demosaicing. We measure the beneficial effect of JPEG block artifact removal on camera identification accuracy. We examine the extent to which JPEG block artifact removal helps circumvention by improving the performance of a denoiser that is used to remove PRNU features. We evaluate the effectiveness of the StirMark watermark remover as a tool for circumventing camera identification, and we examine the similarities and differences between watermark circumvention and camera identification circumvention.

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. 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[2]. 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. Our model does not assume that the photographer is an expert in signal processing. We have chosen our threat model to reflect 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 demosiacing 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 distance function is then invoked to determine the similarity of that image's noise pattern estimate to the each of the cameras' fingerprints.

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 typically assume a model for the noiseless image and a model for the noise, and attempt to separate these two components by maximizing probability. 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 two 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.

3.1 Difference Between Denoisers

A highly simplified abstract model of denoising is that it removes the components of the signal that lie within a subspace, referred to as the *noise space*, spanned by a set of noise basis vectors. Denoising can be thought of as the operation of projecting the signal onto each of these noise basis vectors, thereby obtaining the noise component of the signal, and subtracting this noise component from the original signal. Continuing with this simplified model, we can assume that different denoisers will have different sets of noise basis vectors that define different noise spaces. Therefore, a signal denoised by denoiser A will still contain noise components in terms of denoiser

B. Real denoisers are not based on this simple linear vector space model, but it is nevertheless useful as a conceptual tool for visualizing the differences between denoisers, and to provide intuition for why camera identification using denoised images is possible.

3.2 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.

3.3 Differing Denoisers

In our preliminary experiments, the attacker's denoiser was a C-language reimplementation[1] of a wavelet-domain denoiser by Selesnick[4]. 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.

In the experiments that are currently underway, we are studying the effects of various publicly available denoisers individually and in combination.

3.4 Identical Denoiser

If we adopt a slightly different threat model, we can assume that the attacker has access to an identical denoiser to that which is used for PRNU estimation in camera identification. In this case we investigate 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.

3.5 JPEG Artifact Removal

JPEG, particularly at high compression levels, introduces block artifacts due to the 8x8 pixel blocks on which it operates. We investigate the effect of these artifacts on denoising when used for camera identification circumvention. Due to the scene-dependent nature of the error that is introduced at block boundaries, JPEG block artifacts are an impairment to PRNU-based camera identification. In our preliminary experiments, confirmed this by observing the effect of JPEG artifact removal on the basic camera identification process. A set of 300 test images was split in half. On one set of 150 we observed the correlation of the extracted PRNU pattern with the camera fingerprint template. The other set of 150 images was first processed with by a JPEG artifact removal algorithm[3]. We measured a 10% increase in average correlation between test image PRNU and the camera fingerprint. This indicates that the JPEG artifact removal gives the attacker an advantage in removing PRNU information from the image. Our current research investigates this effect experimentally and analytically.

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

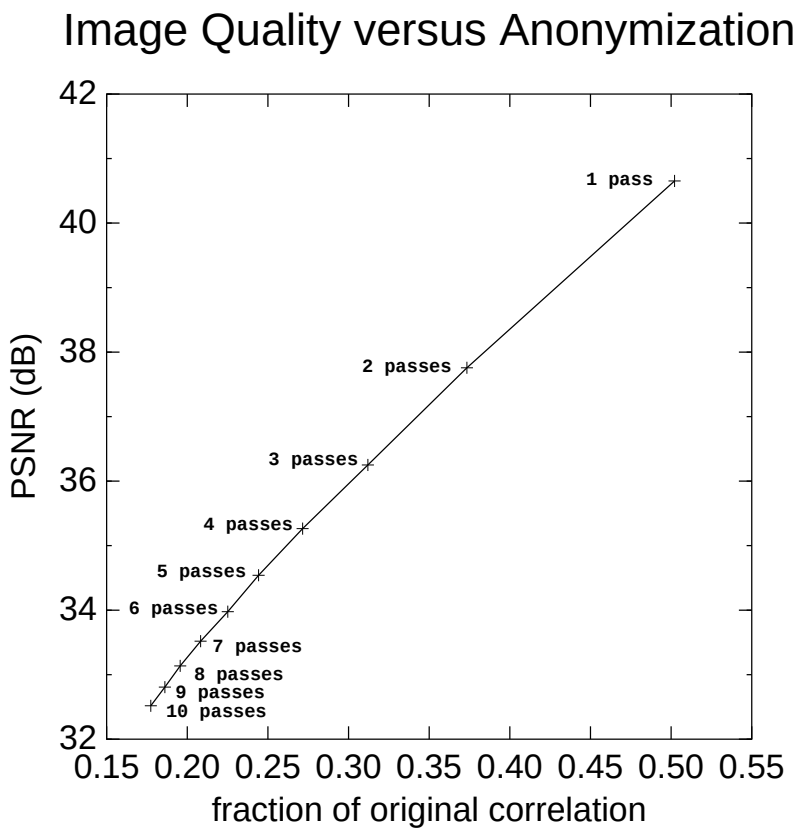

Figure 1: Multiple passes of wavelet-domain denoising are performed on a set of images. The average reduction in correlation is observed between the PRNU of each image and the fingerprint pattern for the camera that produced the image. A significant reduction in correlation is available before ruining the PSNR of the image.

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.

Recompression is the image processing operation of decompressing an image, possibly modifying the uncompressed image, and compressing 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 assume that the goals 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 study DCT- and wavelet-based compression. In the full version of this report, we will also investigate the sequence of the destructive operations.

5 Demosaicing

Another image processing operation that is available in photo editing software that runs on personal computers 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. 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 demosiacing 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.

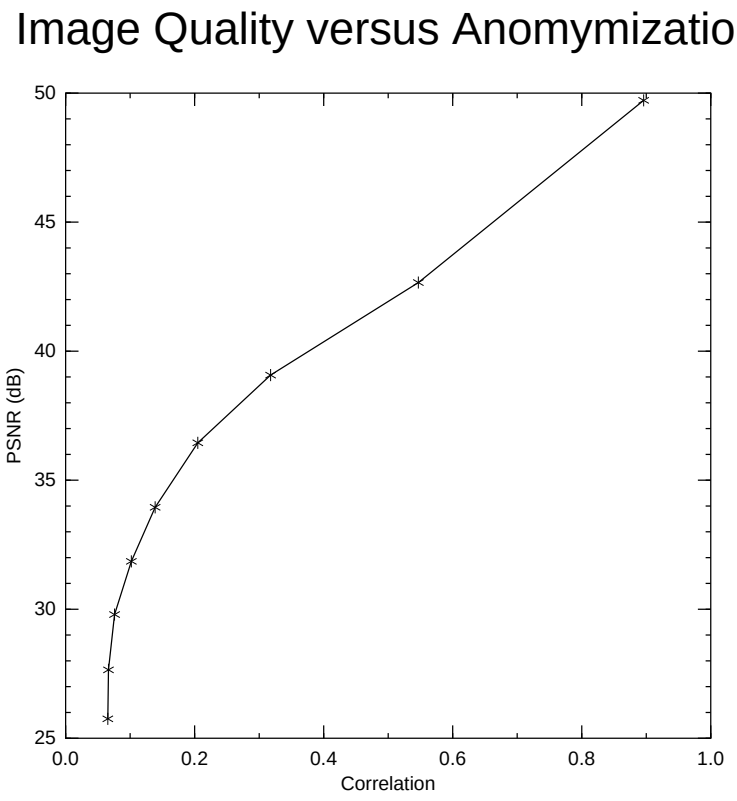

Figure 2: Multiple passes of lossy compression and decompression are performed and its effect on the average correlation between the PRNU pattern of the image and the camera fingerprint is observed. A significant reduction in correlation is obtainable without spoiling the PSNR of the image.

6 Current Work

Our current research continues the investigation described in this abstract and extends it experimentally and analytically. We apply the StirMark operations (scaling/cropping, geometric distortion, column deletion/repeating, etc.) and observe their effectiveness in terms of impairing camera identification. Our goal is to provide useful insight into camera identification circumvention techniques so that this threat can be accurately assessed by users and developers of camera identification technology.

References

[1] Onur G. Guleryuz. Source code for 2d wavelets, http://eeweb.poly.edu/ onur/source.html. Technical report.

[2] Jan Luks, Jessica Fridrich, and Miroslav Goljan. Digital camera identification from sensor noise. In *IEEE Transactions on Information Security and Forensics*, 2006.

[3] Aria Nostratinia. Enhancement of jpeg-compressed images by re-application of jpeg. In *Journal of VLSI Signal Processing*, vol. 27, pp. 69-79, 2001.

[4] Levent Sendur and Ivan W. Selesnick. Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency. In *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, November 2002.