https://doi.org/10.2352/ISSN.2470-1173.2019.5.MWSF-545

Tackling In-Camera Downsizing for Reliable Camera ID Verification

Sinan E. Tandoğan, Enes Altınışık, Salim Sarimurat, Hüsrev T. Sencar
TOBB ETU, Ankara, TURKEY

Abstract

The photo-response non-uniformity (PRNU) of an imaging sensor can be regarded as a biometric identifier unique to each camera. This modality is referred to as camera ID. The underlying process for estimating and matching camera IDs is now well established, and its robustness has been studied under a variety of processing. However, the effect of in-camera downsizing on camera ID verification has not yet been methodologically addressed. In this work, we investigate limitations imposed by built-in camera downsizing methods and tackle the question of how to obtain a camera ID so that attribution is possible with lower resolution media. For this purpose, we developed an application that gathers photos and videos at all supported resolutions by controlling camera settings. Analysis of media obtained from 21 smartphone and tablet cameras shows that downsizing of photos by a factor of 4 or higher suppresses PRNU pattern significantly. On the contrary, it is observed that source of unstabilized videos can be verified quite reliably at almost all resolutions. We combined our observations in a camera ID verification procedure considering downsized media.

Introduction

Characteristics inherent to photo response non-uniformity (PRNU) of a digital imaging sensor, such as randomness, stability, and robustness to common processing, enable it to be used as a unique biometric identifier of a camera. This modality is often referred to as camera ID. The reliability of camera ID in source camera attribution is now well tested, and the development of effective procedures for estimation of PRNU pattern from photographs and videos has been the subject of past and ongoing research [1] [2].

In practice, this capability is widely deployed in a verification setting where the camera ID of a known camera is tested against media captured by an unknown source. In this setting, a camera ID is extracted from a set of photos or videos acquired by the camera, and its match is evaluated with the PRNU pattern extracted from the photo or video in question [3]. In this scenario, it is commonly assumed that extracting a camera ID from a relatively large set of media is sufficient to reliably capture the PRNU profile of a sensor. Motivated by the fact that a typical camera offers a variety of in-camera processing options, we question the validity of this assumption and investigate how camera ID estimation has to be performed to address the source camera attribution problem.

Smartphones and tablets are by far the most common form of camera used today [4]. Not only these devices feature multiple built-in cameras but they also offer significant computational power to capture better quality photos and videos. In any camera, raw image data captured by the sensor is processed in steps through the imaging pipeline before the final image is created. The advancements in smartphone and tablet camera technology adds very sophisticated processing capabilities to this pipeline. Some of these processing are very critical in the context of source attribution as they involve operations that are disruptive to PRNU pattern estimation and matching. Among these, one of the most notable is the downsizing operation which refers to how in-camera processing generates photos and videos at different resolutions.

Cameras capture images and videos at a variety of resolutions. This is typically performed to accommodate different viewing options and support different print qualities. However, downsizing becomes necessary when taking a burst of shots or capturing a video in order to reduce the amount of data that needs processing. Further, to perform complex operations like image stabilization, typically, the sensor is cropped and only the center portion is converted into an image.

The downsizing operation can be realized through different mechanisms. In the most common case, cameras capture data at full sensor resolution which is converted into an image and then resampled in software to the target resolution. An alternative or precursor to sub-sampling is on-sensor binning or averaging which effectively obtains larger pixels. This may be performed either at the hardware level during acquisition by electrically connecting neighboring pixels or by averaging pixel values digitally right after analog-to-digital conversion. Hence as a result of performing downsizing, pixel binning, and sensor cropping, photos and videos captured at different resolutions are likely to yield non-matching camera IDs.

Downscaling operation should ideally preserve important image features. There is, however, no universal definition of what is important in an image and should be preserved and what is unimportant and can be discarded. Therefore, there is no universally accepted way of scaling down an image, and different types of images must be handled differently in order to preserve as much visual content as possible. Conventional image downsampling methods which convolve pixel data with an interpolation function introduce aliasing type visual artifacts. Depending on the choice of interpolation kernel, these methods essentially offer a trade-off between preserving sharpness and smoothing of image features at the expense of higher susceptibility to noise.

Recently, content-adaptive downsizing methods have drawn much attention due to better perceived image quality they offer [5], [6], [7]. Some of these methods are not scaling algorithms per se, and they are mostly used for changing the aspect ratio of an image without making it appear stretched. In essence, these methods identify regions of important image features and perform downsizing to preserve those features. In any case, downsizing is expected to have a detrimental effect on the camera ID verification process, with content-adaptive ones inducing the worst effect, due to their content dependent processing. Since downsizing is an important aspect of image quality, it is safe to assume camera makers will continue to strive for better solutions with further complications to source camera attribution.

In the rest of this paper, we tackle the question of how camera ID of a given camera should be extracted to ensure reliable source verification or identification later. The impact of resampling algorithms on the accuracy of camera ID matching is examined in the following section. Since data driven approaches that involve collecting photos and videos from open sources cannot guarantee coverage of the whole spectrum of possibilities, a camera application is developed for Android mobile operating system to automatically obtain samples directly at different resolutions. Our findings on 21 cameras and results of tests performed on media obtained by these cameras are presented in the subsequent sections. We conclude our paper with a sketch of camera ID estimation process considering downsized media.

Effect of Downsizing on Camera ID Matching

To ensure image quality, downsizing methods focus on different properties such as sharpness, smoothness, and recognizability of important features in the downsized image. For downscaling, sinc filter family interpolation functions, like Lanczos interpolation, are the best choice. In practice, bilinear and bicubic interpolation are more commonly used due to their computational efficiency despite being known for leading to oversmoothing. When downscaling by a factor higher than 2, however, aliasing artifacts start to appear with all methods, effectively causing them to add patterns to the downsized image which are not present in the high resolution version. A common approach to counter this problem is to first blur the image, to remove all details with spatial resolution above Nyquist frequency, and then perform downsizing. This is followed by an optional sharpening to regain sharpness. In contrast, content adaptive methods operate on the principle of maintaining the quality of perceptually important details which may introduce variety of subtle geometric distortions to image.

The operations involved in downscaling inevitably interfere with camera ID verification process. The PRNU pattern is essentially extracted by means of a denoising method whose performance depend on how well local variances of transform coefficients are estimated. In this context, the additional smoothing introduced by a pre-blur filter and the data loss incurred by dimension reduction will lead to a poor estimation of the sensor's PRNU. A post-sharpening operation on the other hand will add spurious noise components. As a result, the match decisions between a camera ID and downsized media will be less reliable.

We performed tests to investigate the effects of downsizing on camera ID verification. For this purpose, we created a test set of 63 images captured by 21 smartphone and tablet cameras (3 per camera) at the highest available resolution. Each camera ID is obtained separately from a set of 50 high-resolution photos and saved for testing. For the tests, 63 high-resolutions images are resized to $\frac{11}{12}, \dots, \frac{1}{12}$ of its original size, yielding 11 downsized versions of each image. For downsizing, we used the well-known bilinear, bicubic, and Lanczos interpolation filters. PRNU patterns extracted from downsized images are matched with the corresponding camera ID which is also downscaled to the same resolution using the same filter. It must be noted that the match between two PRNU patterns (or camera IDs) is evaluated in terms of the peak to correlation energy (PCE) metric. (We refer the reader to [8] for the details on matching process in the interest of limited space.) Fig. 1 shows average PCE values obtained at each scaling factor for the three filters. As can be seen, for all resizing factors higher than $\frac{1}{12}$, computed PCE values are higher than the typical threshold value of 60 used for deciding a match. (Note that this level of downscaling reduces number of pixels in the original image by a factor of 12^2.) We deduce that, downsizing with conventional methods do not impair camera ID verification even under significant data loss and that Lanczos interpolator peforms best at lower scale factors as expected.

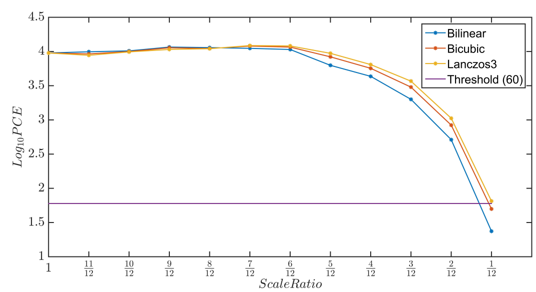

Figure 1. *Average PCE values computed between camera IDs and matching images downsized using three interpolators at varying factors.*

Next, to simulate the case where images are downscaled with an unknown method, we repeated the same test using different interpolation filters to downscale camera ID and images. In this test, we also considered two more downscaling methods. First is the content-adaptive method implemented within the ImageMagick tool. The second one is the built-in downscaling method of the Casper VIA V10 smartphone camera. For the latter case, we gathered 3 images at all supported resolutions, obtained camera ID independently, and performed matching in a similar manner with no knowledge on the details of the downscaling method. In this test, all camera IDs were downscaled using Lanczos interpolator as it yielded the best performance earlier. Fig. 2 presents the change in average PCE at different downscaling factors. Results show that when there is a mismatch between the method used for downscaling the camera ID and the downscaling method applied to test images, performance drops faster. Most critically, for the content-adaptive and the built-in camera downsizing methods, any scaling factor less than $\frac{1}{4}$ impairs source verification.

Determining Sensor Downsizing Behavior

The algorithm for how a camera performs downsizing of full-frame image sensor output when capturing a photo or video is implemented as part of the in-camera processing pipeline and is proprietary to every camera maker. To better understand how different camera makers realize this, we developed an application for Android mobile OS that interacts with the smartphone and tablet cameras and captures photos and videos in a controlled manner. This type of an approach is necessary for two reasons. First, use of the default interface, like the native camera application, to capture photos and videos may only expose certain default settings for observation, thereby not revealing all possible options that may potentially be selected by applications that utilize the built-in camera. The widely accepted scenario where a set of photos or videos captured by a given camera are used for camera ID verification and identification suffers from this limitation. Second, it provides control over camera settings that may have detrimental effects on camera ID verification such as application of electronic zoom, deployment of video stabilization, or use of high dynamic range (HDR) imaging.

Starting with Android version 5.0 (Lollipop), applications can access to device's camera functionality through a system provided standard interface, called Camera2 application programming interface (API). In other words, each application that wants to access one of the built-in system cameras will use the Camera2 API to configure a camera profile by setting its controls. In addition, Camera2 API provides applications the capability to determine which features are present in the camera. Essentially, our application collects data by capturing photographs and videos at all frame resolutions supported by the camera by controlling camera settings.

When the application runs, it configures the rear camera for use and requires the user to move the camera through a well-lit scene to acquire data. In the first phase, application identifies all supported resolutions, turns off auto focus and HDR, and captures 50 photos at the highest resolution and three photos at all other resolutions at camera's default JPEG quality factor varying in the range 85-95. Similarly, during the second phase, it determines all the supported video frame resolutions, removes stabilization, and records 4 seconds long videos at 50 Mbps bitrate to alleviate the effects of video compression. High resolution photos are later used to obtain a camera ID to test the accuracy of match with lower resolution photos. Finally, the native camera application is started to take 50 photos and to record a video for comparison to ensure native application is not performing further processing. A photo is also taken while recording the video to determine active pixel area. This led to the finding that each photo is taken at the highest resolution of the same aspect ratio, which possibly indicates that frames are downscaled by a fixed factor before encoded into a video.

We installed our application on 21 smartphone and tablet models. Tables 1 and 2 present obtained information on supported resolutions with most commonly observed aspect ratios for these

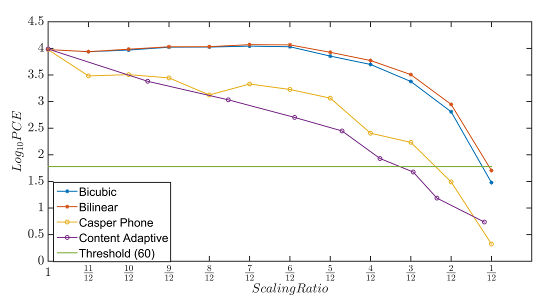

Figure 2. *Average PCE values computed between camera IDs and matching images where camera ID is downsized using Lanczos filter and image using four different downsizing methods at varying factors.*

cameras. It can be seen that cameras can capture photos at $15-30$ resolutions and videos at $10-20$ resolutions, with newer cameras offering more options. Most of these resolutions are in 4:3 and 16:9 aspect ratios. Our observations also show that Android's native camera application support only a few (3-4) of the available resolutions. We were also able to determine that despite using the same sensor (Sony's IMX135 CMOS sensor) Samsung A5 and Samsung Note 3 cameras offer media at different resolutions, confirming our intuition that downsizing is a process performed at the imaging pipeline. We also examined how popular photo and video sharing applications, such as Whatsapp, Instagram, Twitter, and Facebook behave on two cameras. We determined in both cases that they utilize the Camera2 API without doing additional resizing, and the resolutions they save media is not one among those used by the native camera application.

Supported Photo Resolutions and Aspect Ratios

Camera Models	Aspect Ratio				Resolution	
	4/3	16/9	1/1	Others	max.	min.
Casper-V10	9	4	0	6	3264x2448	176x144
GM5Plusd	10	5	1	6	4160x3120	176x144
GMPlus5d	13	5	1	4	4160x3120	176x144
GT-I9500	15	10	0	6	4128x3096	320x180
HTC-OneM9	9	3	5	20	5376x3752	176x144
LG-H815	12	8	0	5	5312x2988	320x240
Lenovo-P1	8	3	0	4	3264x2448	320x240
Lenovo-S90	13	3	0	6	4160x3120	160x120
SM-A300H	6	5	0	2	3264x2448	640x480
SM-A500F	7	5	1	0	4128x3096	320x240
SM-A700F	6	5	1	0	4128x3096	320x240
SM-G361H	4	0	1	1	2576x1932	640x480
SM-G900FQ	6	5	0	1	5312x2988	320x240
SM-G920F	4	5	0	0	3264x2448	256x144
SM-G930F	9	7	5	3	4032x3024	320x240
SM-G960F	9	8	5	3	4032x3024	320x240
SM-J500F	7	5	1	0	4128x3096	320x240
SM-J700F	7	5	1	0	4128x3096	320x240
SM-N9005	6	5	0	0	4128x3096	640x480
SM-T287	4	1	1	3	2576x1932	640x480
Venus	8	2	0	4	3264x2448	160x120

In-Camera Downsizing Effects

We investigated the reliability of using camera IDs obtained at specific resolutions to identifylower resolution media from the same source camera. When capturing media at lower than full-frame resolution, cameras deploy downscaling and cropping as mechanisms for resolution reduction. Cropping reduces the number of pixels, and its application can be detected by the reduction in the field of view of an image or video. In terms of matching performance, use of lesser pixels causes a decrease in the PCE metric. In contrast, downsizing or merging pixels by binning reduces image quality, thereby, causing a weakening in the PRNU pattern. Camera makers utilize both of these operations in proprietary ways in their pipeline. To determine a general approach, we utilize the media acquired by our application from different cameras.

Supported Video Frame Resolutions and Aspect Ratios

Camera Models	Aspect Ratio 4/3	16/9	1/1	Others	Resolution max.	min.
Casper-V10	6	2	0	5	1280x960	160x120
GM5Plusd	8	3	1	6	1440x1080	176x144
GMPlus5d	8	3	1	7	1440x1080	176x144
GT-I9500	5	3	0	7	1440x1080	176x144
HTC-OneM9	3	3	2	7	1920x1088	176x144
LG-H815	7	4	0	4	2048x1536	176x144
Lenovo-P1	3	2	0	7	1440x1080	176x144
Lenovo-S90	5	0	0	2	1440x1080	160x120
SM-A300H	4	2	0	6	1440x1080	176x144
SM-A500F	4	1	1	4	1440x1080	176x144
SM-A700F	4	1	2	4	1440x1080	320x240
SM-G361H	2	1	1	2	1280x960	176x144
SM-G900FQ	4	2	0	5	1920x1080	176x144
SM-G920F	3	5	2	2	3264x1836	176x144
SM-G930F	3	3	3	1	2160x2160	176x144
SM-G960F	5	5	3	3	2160x2160	176x144
SM-J500F	4	1	1	7	1440x1080	176x144
SM-J700F	3	0	1	4	960x720	176x144
SM-N9005	4	1	0	4	1440x1080	176x144
SM-T287	4	1	1	3	1280x960	176x144
Venus	8	2	0	4	1280x960	160x120

Camera ID Verification in Downsized Photos

Camera Models	Scaling Ratio $1-1/2$		$1/2-1/4$		$< 1/4$	
Match(M) or Nonmatch(N)	M	N	M	N	M	N
Casper-V10	7	0	5	0	1	6
GM5Plusd	2	3	3	4	0	10
GMPlus5d	4	3	2	5	0	9
GT-I9500	8	3	2	6	0	12
HTC-OneM9	21	2	2	4	0	8
LG-H815	9	0	3	3	0	10
Lenovo-P1	8	0	4	1	0	2
Lenovo-S90	7	0	2	4	0	9
SM-A300H	9	0	3	0	0	12
SM-A500F	5	0	3	0	0	1
SM-A700F	5	0	4	0	1	2
SM-G361H	4	1	0	0	0	1
SM-G900FQ	2	3	2	1	0	3
SM-G920F	4	0	2	0	0	3
SM-G930F	9	2	0	6	0	7
SM-G960F	12	0	6	0	3	4
SM-J500F	5	0	6	0	0	2
SM-J700F	5	0	5	0	1	2
SM-N9005	5	0	5	0	0	1
SM-T287	1	3	0	0	0	0
Venus	4	0	5	0	0	8

Matching Across Photos

To determine the amount of scaling and cropping photos at lower resolutions undergo, we take the following steps. First, a camera ID is generated using photos at the highest resolution. After ensuring a photo has the correct orientation as the camera ID, a search is performed for the applied scale ratio and cropping amount. Denoting the width and height pair of a camera ID with w_c and h_c and those of photos with w_p and h_p, the camera ID is scaled down by a factor that is equal to the maximum of $\frac{w_p}{w_c}$ and $\frac{h_p}{h_c}$. If the aspect ratio is the same, both downsized camera ID and the photo being verified end up having the same dimensions; otherwise, only one dimension of them will match after scaling. The PRNU pattern is extracted from the photo and searched within the center portion of camera ID by computing normalized cross correlation (NCC). At the shift location that yields the highest NCC value, PCE is computed to verify the match. If the PCE value is found to be lower than the preset threshold value of 60, we assume cropping might have happened at both horizontal and vertical directions. To test for this, camera ID is scaled down in decrements of 0.01 from its original size and the search is repeated at all steps until a reliable match is detected, as performed in [8] and [10] to determine downsizing parameters. Since we have three photos at each resolution, identified cropping and scaling factors are cross checked with all photos at that resolution.

Table 3 shows our findings using photos acquired by our application from the 21 smartphone and tablet cameras. It can be seen that when scaling ratio is above $\frac{1}{4}$ camera ID matches to lower resolution photos in most cases. Our analysis of the mismatching photos revealed that they were either blurry or had dark regions. Since the application captures data automatically at very short intervals, photos captured when the camera was moved too fast or during transitioning from one scene to another had unsuitable content. However, for scaling factors less than $\frac{1}{4}$, except for a very few resolutions, camera ID does not match with very low resolution photos. In all matching cases, we observed that when the aspect ratio of the photo and the camera ID are the same, cameras just perform scaling. If aspect ratio is not the same, we determined that in most cases, the highest resolution photos at 4:3 aspect ratio is scaled down by a factor to match one dimension (maximum of $\frac{w_p}{w_c}$ and $\frac{h_p}{h_c}$) and cropping content equally on both sides of the other dimension.

We investigated two more scenarios to determine whether matching performance can be further improved. In the first case, we took it to an extreme and generated one camera ID for each available resolution. For this, we picked 12 cameras and modified our application to capture 50 photos at all resolutions and re-evaluated the match on the earlier generated photo set. Results show that for 9 of the cameras all camera IDs at $\frac{1}{4}$th of full-frame resolution and below could not be matched to photos at the corresponding resolution. For the remaining 3 cameras, camera ID verification could be sustained until the scaling factor of $\frac{1}{8}$. In the second scenario, rather than downsclaing the high resolution camera ID, we first downscaled the high resolution photos and then obtained a camera ID. We repeated the tests given in Table 3 for 3 of the cameras and observed that neither matching performance nor measured PCE values improve.

Matching Across Videos

Same methodology is followed for videos. The most significant advantage of working with videos is that, unlike in photos, it is possible to extract a camera ID from even the shortest of videos. In contrast, however, the maximum resolution of a video is much smaller than the full-sensor size. This implies larger amount of cropping and scaling has to be performed, which potentially translates to lower PCE values.

In a manner similar to photos, a camera ID is extracted from the highest resolution video and compared with camera IDs extracted from lower resolution videos. In all cases, when extracting a camera ID we removed the loop filtering step at the decoder to better combat video compression [2] To determine how scaling and cropping are performed, for all resolutions, high resolution camera ID is downscaled and a search is performed to determine cropping pattern by computing NCC between the two camera IDs. If no match is found, then high resolution camera ID is downscaled in decrements and a search is performed to determine the amount of cropping.

In our analysis we noticed that the highest resolution video captured by some of the cameras do not utilize the whole sensor area. To exemplify, when the maximum video resolution is in the 16:9 aspect ratio and the corresponding camera ID is matched with a low resolution camera ID in 4:3 aspect ratio, we determined that camera ID extracted from highest resolution video is contained within that of the low resolution video. To better explain this consider an imaging sensor of dimensions 4096×3072 pixels. Downsizing the full-sensor frame by a factor of 4 yields a video frame of size 1024×748 pixels and preserves the 4:3 aspect ratio. Alternatively, first cropping the full-sensor frame to the size of 4096×2304 pixels followed by downsizing by a factor of 2 will yield a frame of size 2048×1152 pixels in 16:9 aspect ratio. Hence, the videos in the former format will yield the full PRNU pattern despite having lower resolution than the latter. So for cameras where there is a mismatch in aspect ratios of highest resolution photo and video, instead of just downscaling the high resolution camera ID to the maximum of $(\frac{w_f}{w_c}, \frac{h_f}{h_c})$ (where w_f and h_f are width and height of the camera ID, respectively) and performing search, we also downscale the high resolution camera ID to minimum of $(\frac{w_f}{w_c}, \frac{h_f}{h_c})$, considering the possibility of the above case, and then perform search.

Our results given in Table 4 show that for most resolutions crop sizes and scaling factors can be determined. In none of the matching cases, we needed to perform a brute-force search of parameters, showing that low-resolution frames are obtained from high resolution ones by scaling first and cropping next when aspect ratio has to change. Our analysis further show that in 15 out of 21 cameras that did not yield a match at lower resolutions, camera IDs still matched until frames were downscaled by a large factor of 10 or higher. To further test this, we extracted two fingerprints from each of those low resolution videos by dividing frames into two groups. It is observed that those two camera IDs also do not match, indicating that downsizing caused suppression of the PRNU pattern. Overall, compared to photos, source verification in downsized videos produces more reliable results as expected. This also indicates that combining PRNU patterns extracted from multiple photos, when possible, will increase the reliability of matching.

Camera ID Verification in Downsized Videos

Camera Models	Scaling Ratio $1-1/4$		$1/4-1/6$		$< 1/6$	
Match(M) or Nonmatch(N)	M	N	M	N	M	N
Casper-V10	1	2	2	4	3	1
GM5Plusd	3	0	3	0	12	0
GMPlus5d	3	0	3	0	13	0
GT-I9500	2	2	2	3	1	5
HTC-OneM9	2	2	1	3	1	6
LG-H815	5	0	4	0	5	1
Lenovo-P1	3	0	5	0	1	3
Lenovo-S90	1	0	2	2	2	0
SM-A300H	1	3	2	3	1	2
SM-A500F	3	0	4	0	2	1
SM-A700F	3	0	4	0	4	0
SM-G361H	5	0	1	0	0	0
SM-G900FQ	2	0	3	0	5	1
SM-G920F	6	2	3	0	1	0
SM-G930F	4	2	4	0	0	0
SM-G960F	4	3	5	0	3	1
SM-J500F	3	0	4	0	5	1
SM-J700F	0	0	3	0	3	2
SM-N9005	1	2	1	1	2	2
SM-N950F	4	3	5	0	1	0
SM-T287	6	0	1	0	1	1
Venus	2	0	7	0	5	0

Cross Media Matching

We also investigated whether the camera ID extracted from highest resolution photos matches those camera IDs obtained from videos at varying resolutions. We followed a similar methodology in determining the scaling ratio and crop size applied to full-sensor frame when recording video. Our results revealed that for some cameras, lower resolution camera IDs were detected with a few pixels shift (3-4) from the center of the high resolution camera ID. To deal with this asymmetry, we have padded 10 rows and columns of pixels around downscaled version of the camera ID, before performing a search. In 15 of 21 cameras, scale ratios and crop sizes are successfully determined. With the remaining 6 cameras the brute force search for these parameters did not yield a successful result. We also determined that for these cameras native camera application is able to capture videos at higher resolution than listed by the Camera2 API, which might be an indication that transition from photo to video involves a more complex operation than we assumed.

Summary

Our analysis shows that when details of the underlying method is known, downsizing does not pose a significant obstacle to camera ID verification. When it is not known, however, downsizing operation results with weakening of the estimated PRNU noise pattern. Our findings on photos and videos captured at various resolutions by 21 cameras show that in-camera downsizing gets disruptive for relatively high downsizing factors. Overall, results suggest that for camera ID verification, one should use the highest resolution photos and videos separately to extract two versions of the camera ID.

In the case of photos, our application and analysis methodology can be used to generate a dictionary of downsizing parameters (*i.e.*, scaling ratio and crop size) determining how each camera model generates photos at lower resolutions. Given a photo with resolution not less than $\frac{1}{4}$th of full-sensor frame, one can perform source verification by downsizing the camera ID accordingly. If the downsizing parameters are not known in advance, the camera ID should be scaled down by a factor that is equal to the maximum of ($\frac{w_p}{w_c}$, $\frac{h_p}{h_c}$) and the PRNU noise has to be searched around the center region of the camera ID by computing normalized correlation at all points.

For lower resolution photos, two approaches should be considered. If there are more than one photos to be source verified, a camera ID should be extracted from these photos rather than performing source verification individually. In the alternative approach, rather than downsizing the camera ID, creation of a camera ID from photos at the corresponding resolution should be considered. Most critically, failure of camera ID verification on lower resolution media should be attributed to inability to reliably extract PRNU noise pattern and not to an actual mismatch of the source.

In the case of videos, one must first ensure that the given video is not stabilized. This can be determined easily by extracting camera IDs from different parts of a video and evaluating their match [11]. When verifying the source of a video, the high resolution camera ID has to be downsized to the size of low-resolution camera ID but a search for downsizing parameters is not necessary. In this case, the match should be determined by resizing the high resolution camera ID by both ratios of $\frac{w_f}{w_c}$ and $\frac{h_f}{h_c}$ and then by searching one camera ID within the other.

When the high resolution camera ID is obtained from photos instead of a video, then the dictionary approach, where downsizing parameters are obtained a-priori, should be taken to perform downsizing properly before verification. Otherwise, the camera ID extracted from the low resolution video must be searched by a brute force approach. For this, the high resolution camera ID is scaled down in decrements of 0.01 from its original size and searched by by computing NCC until a match is detected.

It must finally be noted that the supported resolution information obtained from camera can also be used to verify whether a given media has been resized by another tool by verifying whether the observed resolution is among the ones listed by the Camera2 API.

Acknowledgement

This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) grant 116E273.

References

[1] M. Chen, J. Fridrich, and M. Goljan, “Digital imaging sensor identification (further study),” in *Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505. International Society for Optics and Photonics, 2007, p. 65050P.

[2] E. Altinisik, K. Tasdemir, and H. T. Sencar, “Extracting PRNU noise from H.264 coded videos,” in *26th European Signal Processing Conference (EUSIPCO)*, Sep., 2018.

[3] 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.

[4] F. Richter, “Infographic: Smartphones (and apple) dominate photography,” Dec 2017. [Online]. Available: https://www.statista.com/chart/12337/most-popular-camerabrands-on-flickr/

[5] J. Kopf, A. Shamir, and P. Peers, “Content-adaptive image downscaling,” *ACM Transactions on Graphics (TOG)*, vol. 32, no. 6, p. 173, 2013.

[6] A. C. Öztireli and M. Gross, “Perceptually based downscaling of images,” *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, p. 77, 2015.

[7] S. Avidan and A. Shamir, “Seam carving for content-aware image resizing,” in *ACM Transactions on graphics (TOG)*, vol. 26, no. 3. ACM, 2007, p. 10.

[8] M. Goljan and J. Fridrich, “Camera identification from cropped and scaled images,” in *Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819. International Society for Optics and Photonics, 2008, p. 68190E.

[9] J. Doe, “Recent progress in digital halftoning ii,” *IS&T, Springfield, VA*, p. 173, 1999.

[10] A. Karakücük, A. E. Dirik, H. T. Sencar, and N. D. Memon, “Recent advances in counter prnu based source attribution and beyond,” in *Media Watermarking, Security, and Forensics 2015*, vol. 9409. International Society for Optics and Photonics, 2015, p. 94090N.

[11] S. Taspinar, M. Mohanty, and N. Memon, “Source camera attribution using stabilized video,” in Information Forensics and Security (WIFS), 2016 IEEE International Workshop on. IEEE, 2016, pp. 1–6.