Chapter 23

IDENTIFYING DIGITAL CAMERAS USING CFA INTERPOLATION

Sevinc Bayram, Husrev Sencar and Nasir Memon

Abstract   In an earlier work [4], we proposed a technique for identifying digital camera models based on trace evidence left by their proprietary interpolation algorithms. This work improves on our previous approach by incorporating methods to better detect interpolation artifacts in smooth image parts. To identify the source camera model of a digital image, new features that can detect traces of low-order interpolation are introduced and used in conjunction with a support vector machine based multiclass classifier. Experimental results are presented for source camera identification from among multiple digital camera models.

Keywords: Image forensics, digital cameras, color filter arrays, interpolation

1.   Introduction

Advances in digital technologies have given birth to very sophisticated and low-cost hardware and software tools that simplify the creation, distribution and modification of digital images. This trend has raised new challenges concerning the integrity and authenticity of digital images. It is no longer possible to take the authenticity of digital images for granted. Image forensics, in this context, is concerned with determining the source and potential authenticity of a digital image.

Although digital watermarking technologies [2] have been introduced to address this problem, their realization requires that a watermark be embedded during the creation of a digital image. Essentially, this necessitates digital cameras to have built-in watermarking capabilities. However, this approach has not been adopted by camera manufacturers. Consequently, alternative approaches must be investigated to determine the origin, veracity and nature of digital images. The problem is further complicated by the requirement that a viable solution should require

minimal prior knowledge about the specific digital camera model that was used and the conditions under which the image was captured (blind image authentication). To our knowledge, there are few, if any, techniques that can achieve these goals.

The primary assumption underlying blind image authentication techniques is that all images produced by a digital camera exhibit certain characteristics that – regardless of the captured scene – are unique to the camera due to its proprietary image formation pipeline. Practically all digital cameras encode the camera model, type, date, time and compression information in the EXIF image header. But this information can be easily modified or removed, and, therefore, cannot be used for authentication.

This paper focuses on the source camera model identification problem by identifying traces of the proprietary interpolation algorithm deployed by digital cameras. In particular, we improve on our earlier work [4] by incorporating new methodologies to capture color filter array (CFA) interpolation artifacts due to low-order interpolation.

The next section discusses current approaches for addressing the image source identification problem. The following section, Section 3, briefly describes the image formation process in digital cameras. Section 4 reviews our earlier work [4] and provides details of the improved approach. Section 5 presents the experimental results, and Section 6 contains our concluding remarks.

2.   Current Solutions

In our prior work [7], we studied the source camera model identification problem by identifying and selectively combining a set of image features based on image quality metrics [3] and higher-order statistics [9] of images. This approach requires the design of a classifier that captures the variations in designated image features from different digital cameras.

Another promising approach was proposed by Lukas, *et al.* [8]. In their work, an imaging sensor's pattern noise is characterized via wavelet-based image denoising. The reference noise pattern for a particular digital camera is obtained by averaging the obtained noise residual over a number of high quality JPEG images captured by the camera. Then, a given image is matched to the camera by correlating the noise pattern of the camera (which is claimed to have captured the image in question) with the individual noise pattern extracted from the image itself.

In more recent work [4], we exploited the fact that most state-of-the-art digital cameras, due to cost considerations, employ a single mosaic

structured color filter array (CFA) rather than different filters for each color component. As a consequence, each pixel in the image has only one color component associated with it, and each digital camera employs a proprietary interpolation algorithm to obtain the missing color values for each pixel.

Our approach [4] was inspired by the technique proposed by Popescu, *et al.* [11], which was developed to detect image tampering. Their rationale was that the process of image tampering often requires an upsampling operation which, in turn, introduces periodic correlations between the image pixels. Popescu, *et al.* designated statistical measures to detect these phenomena. In [4], we applied variants of these measures to characterize interpolation algorithms deployed by digital camera models.

The technique presented in this paper improves on our approach in [4] by designating new features. Because of perceptual image quality considerations, designers must tailor the interpolation algorithm to deal with different image features (edges, texture, etc.). This requires the introduction of strong nonlinearities to the interpolation algorithm. However, for relatively smooth image parts, most well-known interpolation algorithms (e.g., bilinear and bicubic methods) ensure satisfactory quality, and more expensive algorithms are not needed. Our premise is that most proprietary algorithms deploy simpler forms of interpolation for smooth image parts. Therefore, traces of the interpolations can be captured more effectively in these portions as opposed to busy image parts where interpolation requires more careful processing. For this purpose, we utilize the results of [6], where the periodicity pattern in the second-order derivative of an interpolated signal is analyzed.

3.   Image Formation in Digital Cameras

The structure and sequence of processing in the image formation pipelines of digital cameras are very similar despite the proprietary nature of the underlying technologies. Light entering the lens of a digital camera is filtered (using anti-aliasing and other filters) and focused on an array of charge-coupled device (CCD) elements, i.e., pixels. The CCD array is the primary and most expensive component of a digital camera. Each light sensing element of the CCD array integrates incident light over the whole spectrum and produces the corresponding electrical signal representation of the scene. Since each CCD element is essentially monochromatic, capturing color images requires separate CCD arrays for each color component. However, due to cost considerations, most digital cameras use only a single CCD array. Different spectral filters,

typically red, green and blue (RGB), are arranged in a pattern so that each CCD element only senses one band of wavelengths. This spectral filter pattern, or mask, is called the color filter array (CFA). The raw image collected from the array is thus a mosaic of red, green and blue pixels.

As each sub-partition of pixels only provides information about a number of green, red and blue pixel values, the missing RGB values for each pixel must be obtained through interpolation (demosaicing). The interpolation is typically carried out by applying a weighting matrix (kernel) to the pixels in the neighborhood of a missing value. Digital camera manufacturers use proprietary demosaicing algorithms that have different kernel size, kernel shape and interpolation functions. Demosaicing is followed by a processing block, which typically involves operations such as color processing and image compression to produce a faithful representation of the scene that was imaged.

Although the image formation pipeline is identical for almost all digital cameras, the exact processing details at all stages vary from one manufacturer to another, and even in different camera models from a single manufacturer. Note that many components in the image formation pipeline, e.g., lenses, optical filters, CCD arrays, are produced by a limited number of manufactures. Because of the overlap, cameras from different manufacturers may exhibit similar qualities, and this should be taken into consideration when associating image features with the digital cameras. However, the interpolation (demosaicing) algorithm and the specific CFA pattern are often unique for each digital camera manufacturer. Our technique, which is described in the next section, exploits variations in color interpolation to classify images taken by different models of digital cameras.

4.   Identifying Interpolation Traces

The methodology proposed by Popescu, *et al.* [11] analyzes traces of up-sampling to identify images (or parts of images) that have undergone resizing; this is accomplished by analyzing the correlation of each pixel value to its neighbors. Since RGB channels are heavily interpolated in a typical digital camera, in [4], we proposed a similar procedure to determine the correlation structure present in each color band and classified images accordingly. Our experimental results showed that the size of the interpolation kernel and the demosaicing algorithm vary from camera to camera [7]. Furthermore, the interpolation operation is highly non-linear, making it strongly dependent on the nature of the depicted scenery. In other words, interpolation algorithms are fine-tuned to pre-

vent visual artifacts. Busy parts of an image have over-smoothed edges or poor color transitions, while smooth parts exhibit linear characteristics. Consequently, we treat smooth and non-smooth portions of images separately in our analysis.

4.1   Non-Smooth Image Parts

We employ the Expectation/Maximization (EM) algorithm to detect traces of interpolation [11]. The EM algorithm has two main steps: an expectation step followed by a maximization step. The expectation value is computed with respect to the unknown underlying variables using the current estimate of the parameters and conditioned on the observations. The maximization step then provides a new estimate of the parameters. These two steps are iterated until convergence [10].

The EM algorithm generates two outputs. One output is a two-dimensional data array called a probability map; each array entry indicates the similarity of an image pixel to one of the two groups of samples (the ones correlated to their neighbors and those that are not) in a selected kernel. Regions of the map identified by the presence of periodic patterns indicate image parts that have undergone up-sampling. The other output is the estimate of the weighting (interpolation) coefficients that designate the contribution of each pixel in the interpolation kernel.
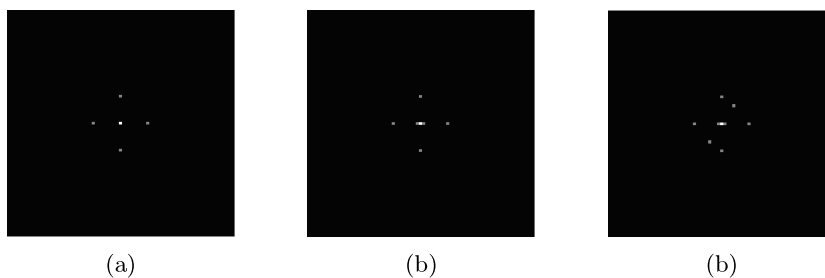

(a)	(b)	(b)

Figure 1. Frequency spectrum of probability maps obtained for (a) Nikon E-2100, (b) Sony DSC-P51, (c) Canon Powershot S200 digital cameras.

Since no *a priori* information is assumed on the size of interpolation kernel (which designates the number of neighboring components used to estimate the value of a missing color component), probability maps are obtained for varying kernel sizes. When observed in the frequency domain, these probability maps yield peaks at different frequencies with varying magnitudes, indicating the structure of the correlation between the spatial samples. Our classifier relies on two sets of features: the set of weighting coefficients obtained from an image, and the peak locations and magnitudes in the frequency spectrum. Figure 1 presents sample

magnitude responses of the frequency spectrum of probability maps for three cameras (Sony, Nikon and Canon). The three responses clearly differ in their peak locations and magnitudes.

4.2   Smooth Image Parts

Gallagher [6] showed that low-order interpolation introduces periodicity in the variance of the second-order derivative of an interpolated signal, which can be used to determine the interpolation rate and algorithm. The interpolation detection algorithm first obtains the second-order derivative of each row and averages it over all rows. When observed in the frequency domain, the locations of the peaks reveal the interpolation rate and the peak magnitudes determine the interpolation method.

We employ a similar methodology to characterize the interpolation rate and the interpolation algorithm employed by a digital camera. Most digital cameras encode and compress images in JPEG format. Due to 8×8 block coding, the DC coefficients may also introduce peaks in the second-order derivative implying the presence of some form of interpolation operation at a rate of 8. Therefore, the peaks due to JPEG compression have to be ignored when attempting to identify interpolation algorithm.

Figure 2 displays the magnitudes of the frequency response for the three models of digital cameras considered in this study. The variations in magnitude indicate that differences exist in the deployed interpolation algorithm. Therefore, the features extracted from each camera include the peak locations (except those due to JPEG compression), their magnitudes, and the energy of each frequency component with respect to other frequency components at all color bands.

5.   Experimental Results

An SVM classifier was used to test the effectiveness of our technique. Several SVM implementations are available; we used the LibSVM package [5]. We also used the sequential forward floating search (SFSS) algorithm to select the best features from a given set of features.

In the first set of experiments, we used the Sony DSC-P51 and Nikon E-2100 camera models. The two cameras both have resolutions of 2 mega-pixels. The pictures are of size 1600×1200 pixels and are obtained with maximum resolution, auto-focus, and other settings at default values. To reduce the dependency on the scenery being viewed, we used pictures of the same scene that were taken by two cameras.
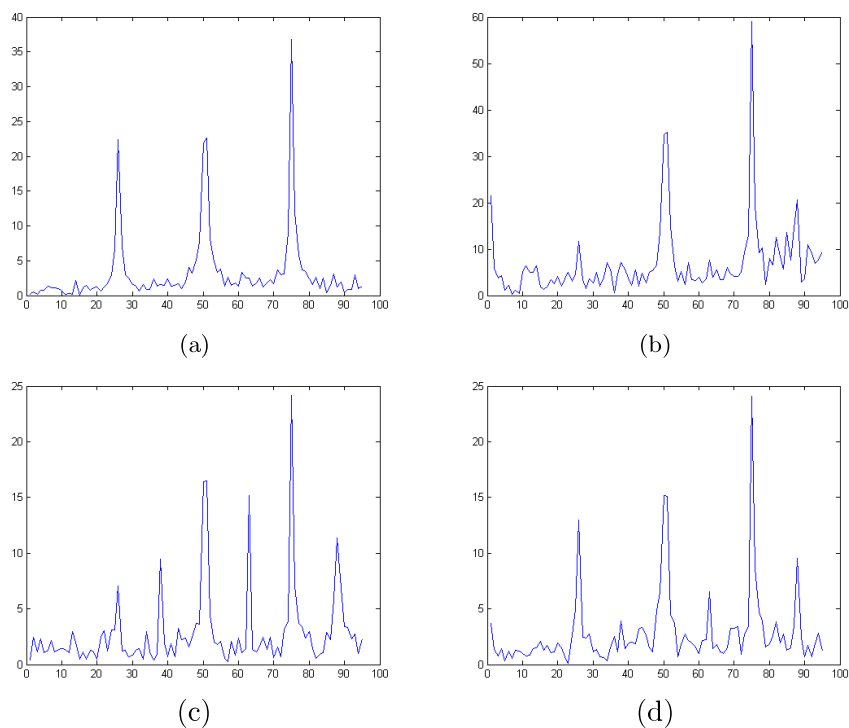
Figure 2. Frequency spectrum of averaged second-order derivatives corresponding to (a) JPEG compression and the three models of digital cameras, (b) Canon Powershot S200, (c) Sony DSC-51, (d) Nikon E-2100 with JPEG output images.

A picture data set was created by capturing 140 images with each camera model. One third of these images were used to train the classifier. The trained classifier was then used to classify the remaining two-thirds of the images. We used 75×75 pixel parts of the images in the experiments. An exhaustive search algorithm was used to partition images into smooth and non-smooth parts based on the variance of each block.

First, we extracted features assuming a 3×3 interpolation kernel for the Sony and Nikon digital cameras. The accuracy was measured as 89.3%. Next, we extracted the features considering the neighboring 4×4 pixels; the correspondingly detection accuracy increased to 92.86%. Finally, the same experiment was repeated for 5×5 neighborhoods, which produced an accuracy of 95.71%.

The three corresponding confusion matrices are presented in Tables 1, 2 and 3, respectively. The data in the tables show that accuracy improves for larger kernel sizes. These results suggest that the actual

size of the interpolation kernel used for CFA interpolation is not smaller than the considered sizes, which was empirically known to be true [7].

Table 1. Confusion matrix for two cameras (3 × 3 interpolation kernel).

		Predicted	
		Nikon	Sony
Actual	Nikon	95.7%	4.3%
	Sony	17.1%	82.9%

Table 2. Confusion matrix for two cameras (4 × 4 interpolation kernel).

		Predicted	
		Nikon	Sony
Actual	Nikon	91.4%	8.6%
	Sony	5.7%	94.3%

Table 3. Confusion matrix for two cameras (5 × 5 interpolation kernel).

		Predicted	
		Nikon	Sony
Actual	Nikon	94.6%	5.4%
	Sony	3.6%	96.4%

Table 4. Confusion matrix for two cameras (periodicity in second-order derivatives).

		Predicted	
		Nikon	Sony
Actual	Nikon	86.9%	13.1%
	Sony	23.3%	76.7%

Similar results were obtained for the smooth image parts using the features based on periodicity in the second-order derivatives. Table 4 shows the accuracy for the two camera case. Note that the latter set of features is not as reliable as the former set of features.

To examine how the proposed features perform for the case of three cameras, we added a Canon Powershot S200 camera to the set of cameras being investigated. The picture set for the Nikon, Sony and Canon

Table 5. Confusion matrix for three cameras (5 × 5 interpolation kernel).

		Predicted		
		Nikon	**Sony**	**Canon**
Actual	**Nikon**	85.7%	10.7%	3.6%
	Sony	10.7%	75.0%	14.3%
	Canon	0.0%	10.7%	89.3%

Table 6. Confusion matrix for three cameras
(periodicity in second-order derivatives).

		Predicted		
		Nikon	**Sony**	**Canon**
Actual	**Nikon**	76.8%	8.9%	14.3%
	Sony	12.5%	76.8%	10.7%
	Canon	19.6%	10.7%	69.6%

cameras included various scenery images downloaded from the Internet. We extracted the features described in Sections 3.1 and 3.2 and used SVM and SFSS to classify the three cameras. An accuracy of 83.33% was obtained when features were extracted from 5 × 5 neighborhoods; the corresponding confusion matrix is provided in Table 5. As shown in Table 6, the accuracy dropped to 74.3% when we attempted to discriminate cameras on the basis of features obtained from smooth image parts.

Table 7. Confusion matrix for three cameras (combined set of features).

		Predicted		
		Nikon	**Sony**	**Canon**
Actual	**Nikon**	94.8%	1.5%	3.7%
	Sony	2.1%	95.3%	2.6%
	Canon	0.0%	2.3%	97.7%

Finally, we combined the two sets of features and repeated the experiment. In this case, the discrimination accuracy increased to 96% for the three camera case as shown in Table 7. The increase in accuracy indicates that the two sets of features capture different characteristics of an image, enabling better identification of the source camera model.

6.   Conclusions

The technique proposed in this paper improves on our previous approach to source camera model identification. To detect traces of color interpolation (artifacts) in RGB color channels, we incorporate several features tuned to capture the periodicity in second-order derivatives from the features obtained using the EM algorithm [4]. A classifier is then designed using the combined set of features and tested to determine the reliability of the selected features in discriminating the source camera model from among two and three cameras. The results are promising; however, the technique is limited to images that are not heavily compressed because compression artifacts suppress and remove spatial correlations between pixels due to CFA interpolation.

References

[1] J. Adams, K. Parulski and K. Spaulding, Color processing in digital cameras, *IEEE Micro*, vol. 18(6), pp. 20-29, 1998.

[2] A. Akansu, E. Delp, T. Kalker, B. Liu, N. Memon, P. Moulin and A. Tewfik, Special Issue on Data Hiding in Digital Media and Secure Content Delivery, *IEEE Transactions on Signal Processing*, vol. 41(6), 2003.

[3] I. Avcibas, N. Memon and B. Sankur, Steganalysis using image quality metrics, *IEEE Transactions on Image Processing*, vol. 12(2), pp. 221-229, 2003.

[4] S. Bayram, H. Sencar, N. Memon and I. Avcibas, Source camera identification based on CFA interpolation, *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, pp. 69-72, 2005.

[5] C. Chang and C. Lin, LibSVM: A library for support vector machines, version 2.81 (www.csie.ntu.edu.tw/~cjlin/libsvm), November 20, 2005.

[6] A. Gallagher, Detection of linear and cubic interpolation in JPEG compressed images, *Proceedings of the Second Canadian Conference on Computer and Robot Vision*, pp. 65-72, 2005.

[7] M. Kharrazi, H. Sencar and N. Memon, Blind source camera identification, *Proceedings of the IEEE International Conference on Image Processing*, pp. 709-712, 2004.

[8] J. Lukas, J. Fridrich and M. Goljan, Determining digital image origin using sensor imperfections, *Proceedings of the SPIE*, vol. 5685, pp. 249-260, 2005.

[9] S. Lyu and H. Farid, Detecting hidden messages using higher-order statistics and support vector machines, *Proceedings of the Information Hiding Workshop*, 2002.

[10] T. Moon, The expectation-maximization algorithm, *IEEE Signal Processing Magazine*, vol. 13, pp. 47-60, November 1996.

[11] A. Popescu and H. Farid, Exposing digital forgeries by detecting traces of re-sampling, *IEEE Transactions on Signal Processing*, vol. 53(2), pp. 758-767, 2005.