IMAGE TAMPER DETECTION BASED ON DEMOSAICING ARTIFACTS

Ahmet Emir Dirik

Polytechnic Institute of NYU
Electrical & Computer Engineering Dept.

Nasir Memon

Polytechnic Institute of NYU
Computer Science & Engineering Dept.

ABSTRACT

In this paper, we introduce tamper detection techniques based on artifacts created by Color Filter Array (CFA) processing in most digital cameras. The techniques are based on computing a single feature and a simple threshold based classifier. The efficacy of the approach was tested over thousands of authentic, tampered, and computer generated images. Experimental results demonstrate reasonably low error rates.

Index Terms— Digital image forensics, tamper detection, CFA

1. INTRODUCTION

This paper focuses on the problem of image tamper detection. We use the term *tamper* in a very broad sense to mean any post-processing operation that has been performed on an image. In the past few years, many image tamper detection techniques have been proposed. Some of these techniques focus on detecting a particular type of tampering operation such as re-compression [1], cloning [2, 3], splicing [4], resizing [5]. We call such techniques *Targeted Tamper Detection* techniques. Another class of techniques try to detect the presence of generic image manipulation operations that may be indicative of tampering such as filtering, down-sampling, up-sampling, compression, rotation, etc. [6, 7, 8, 9]. These techniques do not necessarily determine what operation has been performed but only that the entire image has been subject to post-processing. We call such techniques *Universal Tamper Detection* techniques. A third category of techniques for local tamper detection search for inconsistencies in image characteristics, statistics and content across different regions and are hence able to detect and localize image tampering. Examples include techniques that detect the presence of inconsistencies in sensor noise pattern [10], chromatic aberration [11], lighting [12], CFA (Color Filter Array) demosaicing artifacts[13, 14]. We call such techniques *Localized Tamper Detection* techniques.

In this paper, we develop Color Filter Array (CFA) demosaicing based tamper detection techniques which can be used to detect both local and global tampering operations. The proposed techniques do not target any specific operation but are applicable to a variety of operations such as splicing, retouching, re-compression, resizing, blurring etc. The proposed methods differ from known universal tamper detection techniques [6, 7, 8] in the sense they do not require a complex classifier; instead they use only one threshold to make a decision about the image in question. The basic approach is based on the fact that typically an image tampering operation alters CFA demosaicing artifacts in a measurable way. The lack of CFA artifacts or the detection of weak CFA artifacts may indicate the presence of global or local tampering. Based on this approach, we propose two methods. One based on *CFA pattern number estimation* and the other based on *CFA based noise analysis*.

The CFA pattern estimation method used in this work was first introduced in our previous work [15] to distinguish real images from computer generated (CG) ones. Here, we apply the method to the tamper detection problem. The second method, CFA based noise analysis, relies on the fact that sensor noise power in CFA interpolated pixels should be significantly lower than non-interpolated pixels due to the low pass nature of CFA demosaicing. In [16], a similar approach was used to capture CFA traces to distinguish CG from real images by high pass filtering and Fourier analysis.

As will be shown in the following sections, the proposed CFA based features can be used to detect both global and local tampering without using a high dimensional classifier. Besides, the proposed method provides better results for detection of resampling and resizing operations than a state-of-the-art digital forensics method introduced in [9].The rest of the paper is organized as follows: The description of the proposed techniques is given in Section 2. Experimental results for local and global tamper detection are presented in Section 3. Finally, conclusions are drawn in Section 4 with a brief discussion on the limitations of the proposed techniques.

2. DESCRIPTION OF CFA BASED FEATURES

In this section we provide details of the CFA processing artifact based features that we compute for image tamper detection. Specifically, we compute two features and develop tamper detecting techniques based on each of these features, as described in the two subsections below.

2.1. Feature 1: CFA pattern number estimation

The first method mainly relies on an estimation procedure for the CFA interpolation pattern of the source digital camera. Here, to identify the CFA pattern of an image, the image is re-interpolated with several candidate CFA patterns. For each of these candidate patterns, the Mean Square Error (MSE) between the input and re-interpolated image is computed. For a 2×2 cell CFA, there are 36 different filter arrangements each potentially being a candidate CFA pattern. However, digital cameras in the market generally use one of the 4 Bayer CFA arrangements (see Fig. 1) out of the 36 possibilities. Therefore, we compute MSE values with just the 4 Bayer patterns as candidates.

The *key observation* that leads to our tamper detection technique is that for an image that is *not* tampered, it is expected that one of the MSE values out of the 4 computed with each candidate pattern should be significantly smaller than the others. Specifically, the MSE computed for the actual CFA pattern used for the image should be much smaller than the other 3 patterns. If none of the 4 MSE values are significantly smaller than the others, the image may have undergone a postprocessing operation which removes the traces of demosaicing. Hence, manipulations such as resizing, recompression, and

filtering can be detected through the analysis of MSE values. In addition, the inconsistency of the estimated CFA patterns computed for different image sub-blocks can provide an indication of a likely local tampering/retouching.

We now describe the technique in more detail. Let $L_c(x, y)$ be the image intensity of color channel c at spatial location (x, y) and $c \in \{R, G, B\}$. Let $\Psi_{k,c}$ represent the set of color filter array locations of the channel c for a particular type of CFA pattern denoted by k. The corresponding color filter mask of $\Psi_{k,c}$ is defined as:

$$\theta_{k,c}(x, y) = \begin{cases} 1, & (x, y) \in \Psi_{k,c} \\ 0, & \text{otherwise} \end{cases}$$

In this paper, we restrict the maximum value of possible CFA arrangements k with 4. These 4 different CFA arrangements are shown in Fig. 1.

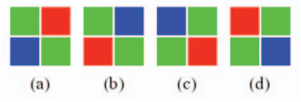


Fig. 1. 4 Different Bayer CFA patterns

To estimate the presence of CFA interpolation artifacts, the image is divided into $W \times W$ sub-blocks and only the non-smooth blocks are used in the computation of CFA feature. We restrict our attention to non-smooth areas as the interpolation of missing pixel values in non-smooth regions is relatively more complex than interpolating pixels in smooth regions. Therefore, re-interpolation errors in non-smooth regions take significantly higher values than the error values computed in smooth regions. Further, for the sake of simplicity we chose the demosaicing algorithm, f, to be bilinear. However, utilizing bicubic interpolation did not change the estimation results significantly.

We denote each non-smooth block as B_i, where $i = 1, ..., N$. N is the number of non-smooth blocks in a given image. The corresponding re-interpolated blocks, using filter k, are denoted with $\hat{B}_{i,k}$. Essentially, $\hat{B}_{i,k}$ is computed as a convolution between the bilinear kernel and the re-sampled block B_i with the kth CFA pattern θ_k. The re-interpolation error of ith sub block for the kth CFA pattern is defined as: $\hat{B}_{i,k} = f(B_i, \theta_k)$, and $k = 1, ..., 4$. The MSE error between the blocks B and $\hat{B}$ is computed in non-smooth regions all over the image as:

$$E_i(k, c) = \frac{1}{W \times W} \sum_{x=1}^{W} \sum_{y=1}^{W} (B_i(x, y, c) - \hat{B}_{i,k}(x, y, c))^2$$

E_i is a 4×3 matrix of mean square errors for each color channel and CFA pattern. To detect the relative error distances between color channels, a new error matrix $E_i^{(2)}$ is created by normalizing all the rows of the E_i, as:

$$E_i^{(2)}(k, c) = 100 \times \frac{E_i(k, c)}{\sum_{l=1}^{3} E_i(k, l)}, \quad c = 1, ..., 3.$$

Due to the lesser number of interpolated pixels in the green channel, the minimum MSE values are observed in the green channel. The green channel column of the normalized error, $V_i(k)$, is used in extraction of our tamper detection feature. The normalized green channel column vector is defined as:

$$V_i(k) = 100 \times \frac{E_i^{(2)}(k, 2)}{\sum_{l=1}^{3} E_i^{(2)}(l, 2)}$$

In the presence of Bayer filter demosaicing, it is expected that one of the CFA pattern estimation errors in V_i vector should be significantly smaller than others. Therefore, the uniformity of the V_i vector can be used as an indicator of possible demosaicing operation. Thus, a new metric to estimate the uniformity of V_i is defined as

$$U(i) = \sum_{l=1}^{4} |V_i(l) - 25|$$

To make an over all decision for the given image, all $U(i)$ values are computed for all non smooth image blocks. Finally, the median value of the U vector is computed as a final CFA trace metric as: $F_1 = median(U)$. The higher the CFA metric, F_1 is, the more likely, the given image is interpolated with CFA demosaicing and did not undergo any significant post processing or tampering.

2.2. Feature 2: CFA based noise analysis

Another way to measure CFA demosaicing artifacts is to look at sensor noise power changes all across the image. If a given image is CFA interpolated, the sensor noise in the interpolated pixels is expected to be suppressed due to the low pass nature of interpolation. As a result, the variance of the sensor noise in interpolated pixels becomes significantly lower than the sensor noise power in non-interpolated pixels after demosaicing. CFA demosaicing artifacts can hence be measured by taking the ratio of noise variances of interpolated and un-interpolated pixels. If this ratio is close to 1, the given image is assumed to be tampered.

To estimate the sensor noise, the dual tree wavelet based denoising algorithm [17] is applied to the green channel of the given image. To separate interpolated pixels from non-interpolated ones, green channel filter mask $\theta_{k,c}$ (Bayer pattern filter $k = 1$ for green channel $c = 2$) is used to determine the positions of two pixel groups (Fig. 1). Using $\theta_{1,2}$, noise residue values of interpolated and non-interpolated pixel sets are placed into two vectors, A_1 and A_2. Finally, the ratio of sensor noise variances are measured as:

$$F_2 = max(\frac{var(A_1)}{var(A_2)}, \frac{var(A_2)}{var(A_1)})$$

where, $var(A_i)$ is the variance of A_i vector and $max(x, y)$ is the function which returns the maximum value of x and y. Feature 2, F_2, is sensitive to the difference of noise variances in interpolated and un-interpolated pixels. If there is not any significant variance difference, the metric converges to one. Otherwise, the metric takes positive values larger than one.

3. EXPERIMENTS

In this section, the efficacy of the proposed features , F_1 and F_2, was tested for image tamper detection over 1000 real images taken from 10 different cameras and over 10000 Computer Generated (CG) images. The camera model and brands used are given in Table 1.

Table 1. Cameras used in the experiments

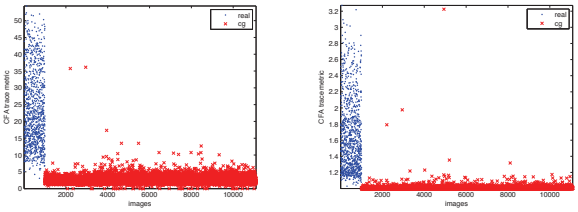
Canon A70	Nikon D50 (dslr)	Canon A80
Konica Z3	Canon EOS (dslr)	Sony H1
Canon S1	Sony P150	
Sony DSC70	Sony DSC90	

Table 2. Tamper detection with F_1 and F_2 features. (TP=true positive, FP=false positive)

	F_1	F_2
blurring	TP 0.999, FP 0.000	TP 0.996, FP 0.000
downsize	TP 0.998, FP 0.001	TP 1.000, FP 0.000
upsize	TP 0.998, FP 0.000	TP 1.000 , FP 0.000
jpeg Q50	TP 0.992, FP 0.002	TP 0.991, FP 0.012
rot 1 deg	TP 0.955, FP 0.031	TP 0.998, FP 0.001

3.1. Detection of CFA traces: Is it CG or Real ?

To evaluate the performance of the two features, F_1 and F_2 on CFA artifact detection, F_1 and F_2 were computed for the real and CG images. The resulting feature plots are shown in Fig. 2. It can be seen that both F_1 and F_2 can be used to distinguish CG and real images with very high accuracy. The false positive rates for F_1 and F_2 were 0.0026 and 0.0027 respectively. The total detection accuracy rates were 0.9963 and 0.9959, respectively.



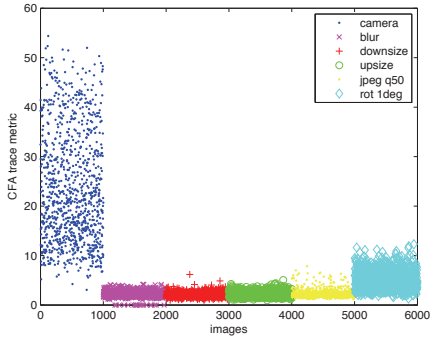
(a) Feature 1 (F_1) (b) Feature 2 (F_2)

Fig. 2. Feature 1 and 2 for real (• symbol) and cg (× symbol) image sets.

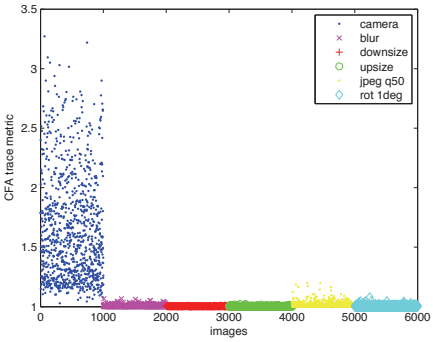
3.2. Tamper detection

The efficacy of proposed features was investigated for several image manipulation techniques such as blurring, image resizing (up and down sizing), JPEG re-compression, and rotation. We conducted experiments over 1000 real images taken from 10 different cameras given in Table 1. Then, 5 different image tampering operations (blurring with 5x5 kernel, 10% downsizing, 10% upsizing, jpeg re-compression with Q50, and 1 degree rotation) were applied to 1000 real images and as a result, 6000 tampered images were created. Finally, the proposed features were computed over 1000 real and 6000 tampered images. The CFA features computed over tampered (labelled with negative) and non-tampered image (labelled with positive) sets are given in Fig. 3. The corresponding TP (true positive) and FP (false positive) rates computed with linear thresholding for different tampering methods are given in Table 2. The thresholds used in the experiments were empirically determined based on the values of F_1 and F_2. These thresholds can be either fixed to a particular value for universal tamper detection or can be fine tuned for detection of targeted tampering.

From the results given in Table 2, it is seen that both F_1 and F_2 can be used to detect different image tampering operations with high accuracies. Especially, F_2 detects image down and upsizing with 100% accuracy for 1000 real and 2000 resized images. Another interesting observation is that F_1 detects CFA traces and outputs high



(a) Tamper detection with feature 1 (F_1)



(b) Tamper detection with feature 2 (F_2)

Fig. 3. Tamper detection with proposed features

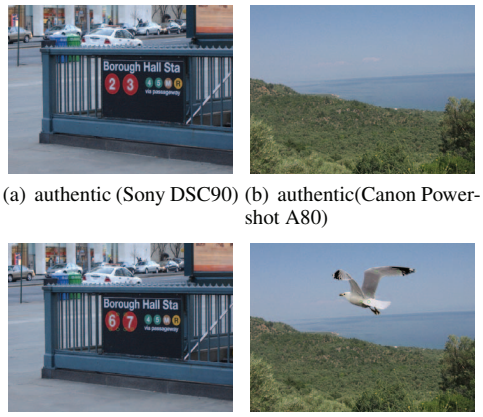
values even if the images are rotated with 1 degree (see Fig. 3.a) where as F_2 converges to one for small degree rotations.

3.3. Local tamper detection

Local image tampering generally distorts image statistics in tampered image region. Therefore, it is expected that tampered image regions should yield different CFA demosaicing artifacts as compared to the rest of the image. To measure this property, two images taken from two different cameras (Sony DSC 90 and Canon Powershot A80) were tampered. The original and tampered images are shown in Fig. 4. Here, we propose two different methods for local tamper detection. The first method is based on the estimation of CFA pattern from the green channel and the second method is based on the noise power analysis in interpolated and non interpolated image pixels.

To test the first detection approach, given image's green channel was divided into $W \times W$ sub blocks and each block was re-interpolated with two different green bayer patterns shown in Fig. 1. The mean square estimation errors for two different patterns were then compared with each other. The CFA pattern which yields minimum square error was assumed as the original CFA pattern of the source camera. The inconsistencies in this CFA pattern were evaluated as an indication of likely tampering in the corresponding image sublock. Detected CFA inconsistencies in tampered images are shown in the second row of the Fig. 4.

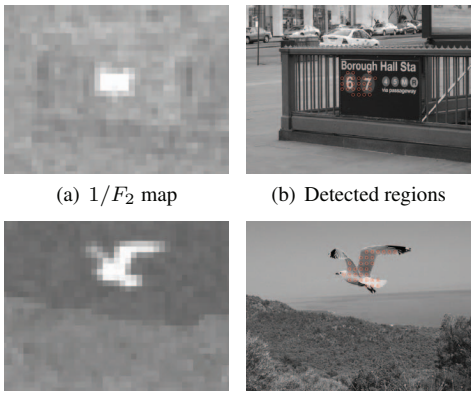
The second approach is based on the feature 2 introduced in Section 2.2. To test this method, tampered images shown in Fig. 4.c and



(a) authentic (Sony DSC90) (b) authentic(Canon Powershot A80)

(c) Tampered image (d) Tampered image

Fig. 4. Tamper detection with CFA pattern estimation.



(a) $1/F_2$ map (b) Detected regions

(c) $1/F_2$ map (d) Detected regions

Fig. 5. Tamper detection with F_2 feature. $1/F_2$ maps are given in the first column.

d were divided into $W \times W$ sub-blocks with $W/2$ steps ($W = 96$). Then, for each image sub-block, F_2 features were computed as introduced in Section 2.2. Finally, computed F_2 features were applied to a linear threshold T. The image sub-blocks yielding lower F_2 values than threshold T were assumed as likely tampered regions. CFA feature maps ($1/F_2$) computed for tampered images and detected tampered regions are shown in Fig. 5.

4. CONCLUSIONS AND LIMITATIONS

In this paper, two different feature based tamper detection methods are introduced. The proposed features do not require any complex machine learning algorithms to make decisions. Instead, the proposed features are used with empirically determined linear thresholds to determine whether given images are tampered or not. The efficacy of the features were tested over thousands of images and 10 different digital cameras. The experimental results show that the proposed features can be used successfully for tamper detection problem with very low error rates.

One limitation of the proposed features is that images taken with X3 Foveon sensors do not exhibit any CFA demosaicing artifacts. Thus, the proposed techniques will not work for images acquired with X3 Foveon sensors. Another limitation is that proposed local tamper detection techniques are sensitive to strong JPEG re-compression and resizing. Since these type of operations distort/suppress CFA artifacts, CFA based local tamper detection may not be successful after these operations. The proposed scheme may also not work well if the tampered region area is too small ($W = 96$ for F_2) or the adversary applies re-demosaicing after tampering. If the adversary does not have prior knowledge about the source camera demosaicing algorithm, the second demosaicing attempt may still be detected.

5. REFERENCES

[1] S. Ye, Q. Sun, and E. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *IEEE International Conference on Multimedia and Expo*, 2007, pp. 12–15.

[2] J. Fridrich, D. Soukal, and J. Lukáš, "Detection of copy-move forgery in digital images," in *Digital Forensics Research Workshop*, August 2003.

[3] A. C. Pospescu and H. Farid, "Exposing digital forgeries by detecting dublicated image regions," 2004.

[4] Y.-F. Hsu and S.-F. Chang, "Image splicing detection using camera response function consistency and automatic segmentation," in *International Conference on Multimedia and Expo*, 2007.

[5] A. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of re-sampling," *IEEE Transactions on Signal Processing*, 2004.

[6] S. Bayram, İ. Avcıbaş, B. Sankur, and N. Memon, "Image manipulation detection with binary similarity measures," in *Proc. of 13th European Signal Processing Conference*, 2005, vol. I, pp. 752–755.

[7] S. Bayram, İ. Avcıbaş, B. Sankur, and N. Memon, "Image manipulation detection," *Journal of Electronic Imaging*, vol. 4, October-December 2006.

[8] H. Gou, A. Swaminathan, and M. Wu, "Noise features for image tampering detection and steganalysis," in *IEEE International Conference on Image Processing*, 2007.

[9] A. Swaminathan, M. Wu, and K. J. Ray Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Transactions on Information Security and Forensics*, vol. 3, pp. 101–117, March 2008.

[10] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Security and Forensics*, vol. 3(1), pp. 74–90, March 2008.

[11] M. K. Johnson and H. Farid, "Exposing digital forgeries through chromatic aberration," in *MM&Sec '06: Proceedings of the 8th workshop on Multimedia and security*, New York, NY, USA, 2006, pp. 48–55, ACM.

[12] M. K. Johnson and H. Farid, "Exposing digital forgeries in complex lighting environments," *IEEE Transactions on Signal Processing*, vol. 2, no. 3, pp. 450–461, 2007.

[13] A. Popescu and H. Farid, "Exposing digital forgeries in color filter array interpolated images," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, pp. 3948–3959, 2005.

[14] M.-C. Poilpré, P. Perrot, and H. Talbot, "Image tampering detection using bayer interpolation and jpeg compression," in *Proc. of the 1st international conference on Forensics applications and techniques in telecommunications, information and multimedia and workshop*, 2008.

[15] A. E. Dirik, S. Bayram, H. T. Sencar, and N. Memon, "New features to identify computer generated images," in *IEEE International Conference on Image Processing, ICIP '07*, 2007, vol. 4, pp. IV–433 – IV–436.

[16] A. C. Gallagher and T. Chen, "Image authentication by detecting traces of demosaicing," in *Proc. CVPR WVU Workshop*, 2008.

[17] L. Sendur and I.W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 438–441, Dec. 2002.