IMAGE MANIPULATION DETECTION WITH BINARY SIMILARITY MEASURES

Sevinç Bayram[a], İsmail Avcıbaş[a], Bülent Sankur[b], Nasir Memon[c]

a Department of Electronics Engineering, Uludağ University, Bursa, Turkey.
b Department of Electrical and Electronics Engineering, Boğaziçi University, İstanbul, Turkey.
c Department of Computer and Information Science, Polytechnic University, Brooklyn, NY, USA.

sevincbayram@yahoo.com, avcibas@uludag.edu.tr, sankur@boun.edu.tr, memon@poly.edu

ABSTRACT

Since extremely powerful technologies are now available to generate and process digital images, there is a concomitant need for developing techniques to distinguish the original images from the altered ones, the genuine ones from the doctored ones. In this paper we focus on this problem and propose a method based on the neighbor bit planes of the image. The basic idea is that, the correlation between the bit planes as well the binary texture characteristics within the bit planes will differ between an original and a doctored image. This change in the intrinsic characteristics of the image can be monitored via the quantal-spatial moments of the bit planes. These so-called Binary Similarity Measures are used as features in classifier design. It has been shown that the linear classifiers based on BSM features can detect with satisfactory reliability most of the image doctoring executed via Photoshop tool.

Keywords: Digital image forensics, image processing, binary similarity measures, classification.

1. INTRODUCTION

The advances in digital technologies have given birth to very sophisticated and low-cost tools that are now integral parts of information processing. This trend brought with it new challenges concerning the integrity and authenticity of digital documents, in particular images. The most challenging of these is that digital images can now be easily created, edited and manipulated without leaving any obvious traces of having been modified. As a consequence, one can no longer take the authenticity of images for granted, especially when it comes to legal photographic evidence. Image forensics, in this context, is concerned with determining the source and potential authenticity of a digital image.

Digital watermarks can serve in a scheme to authenticate images. However, presently the overwhelming majority of images that circulate in the media and Internet do not contain a digital watermark. Hence in the absence of widespread adoption of digital watermarks or concurrently with it, we believe it is necessary to develop image forensic techniques. We define image forensics as the art of reconstituting the set of processing operations, called overall doctoring, that the image has been subjected to. In turn these techniques will enable us to make statements about the origin, veracity and nature of digital images.

In a prior work [6], we studied the same problem of reliably discriminating between "doctored" images (images which are altered in order to deceive people) from untampered original ones. The detection scheme was based on training a classifier based on certain image quality features, called also "generalized moments". Scaling, rotation, brightness adjustment, blurring, enhancement etc. or some particular combinations of them are typical examples of doctoring. A frequent image manipulation involves the pasting of another image, skillfully manipulated so to avoid any suspicion. Since the image manipulations can be very subtle to eschew detection, the discriminating features can be easily overwhelmed by the variation in the image content. It is, thus, very desirable to obtain features that remain independent of the image content, so that they would only reflect the presence, if any, of image manipulations.

2. BINARY SIMILARITY MEASURES

We assume that altering an image changes the correlation between and within bit planes. Therefore the quantal-spatial correlation between the bit planes of the original image will differ from that of the bit planes of the doctored images. Consequently certain statistical features extracted from the bit planes of images can be instrumental in revealing the presence of image manipulations. Since each bit plane is also a binary image, we start by considering similarity measures between two binary images. These measures, called Binary Similarity Measures (BSM) were previously employed in the context of image steganalysis.[1, 3]. In this paper we measure the correlation between bit planes numbered 3-4, 4-5, 5-6, 6-7 and 7-8 for the red channel and bit planes 5-5 of the red and blue channels.

Classical measures are based on the bit-by-bit matching between the corresponding pixel positions of the two images. Typically, such measures are obtained from the scores based on a contingency table (or matrix of agreement) summed over all the pixels in an image. In this study, we have found that it is more relevant to make comparison based on *binary texture statistics*. Let $\mathbf{x}_i = \{x_{i-k}|,\ k = 1,\ldots,K\}$ and $\mathbf{y}_i = \{y_{i-k}|,\ k = 1,\ldots,K\}$ be the

sequences of bits representing the K-neighborhood pixels, where the index i runs over all the MxN image pixels. For K=4 we obtain the four stencil neighbors and for K=8 we obtain the 8 neighbors. Let

$$c_{r,s} = \begin{cases} 1 & if\ x_r = 0 \quad and \quad x_s = 0 \\ 2 & if\ x_r = 0 \quad and \quad x_s = 1 \\ 3 & if\ x_r = 1 \quad and \quad x_s = 0 \\ 4 & if\ x_r = 1 \quad and \quad x_s = 1 \end{cases} \tag{1}$$

Then we can define the agreement variable for the pixel x_i as: $\alpha_i^j = \sum_{k=1}^{K} d(c_{i,j-k}, j), \quad j = 1 \ldots 4, \quad K = 4$, where

$$\delta(m,n) = \begin{cases} 1 & , \quad m = n \\ 0 & , \quad m \neq n \end{cases}. \tag{2}$$

The accumulated agreements can be defined as:

$$a = \frac{1}{MN}\sum_i \alpha_i^1 \ , \quad b = \frac{1}{MN}\sum_i \alpha_i^2 \ ,$$
$$c = \frac{1}{MN}\sum_i \alpha_i^3 \ , \quad d = \frac{1}{MN}\sum_i \alpha_i^4 \ . \tag{3}$$

These four variables {a,b,c,d} can be interpreted as the one-step co-occurrence values of the binary images. Obviously these co-occurrences are defined for a specific bit plane b, though the bit plane parameter was not shown for the sake simplicity. Normalizing the histograms of the agreement scores for the b[th] bit-plane (where now $a_i^j = a_i^j(b)$) one obtains for the j'th co-occurrence:

$$p_j^\beta = \sum_i \alpha_i^j \Big/ \sum_i \sum_j \alpha_i^j \ ; \quad \beta = 3 \ldots 8 \tag{4}$$

In addition to these we calculate the Ojala [4] texture measures as follows. For each binary image on the b[th] bit-plane we obtain a 256-bin histogram based on the weighted K=8 neighborhood as in Fig. 1. For each 8-neighborhood pattern, the histogram bin numbered $n = \sum_{k=0}^{7} x_{i-k} 2^i$ is augmented by one.

1	2	4
128	x_i	8
64	32	16

(a)

0	1	0
1	x_i	0
0	1	1

(b)

Fig. 1 (a) The weighting of the neighbors in the computation of Ojala score. (b) An example: Ojala score S=2+16+32+128=178

Let the two normalized histograms be denoted as S_n^b, $n = 0 \ldots 255$ and $b = 3..7$. The resulting Ojala measure is the mutual entropy between the two distributions belonging to adjacent planes *b* and *b+1*:

$$m_b = -\sum_{n=1}^{N} S_n^b \log S_n^{b+1} \ . \tag{5}$$

Table I. Binary Similarity Measures

Similarity Measure	*Description*
Sokal & Sneath Similarity Measure 1	$m_1 = \frac{a}{a+b} + \frac{a}{a+c} + \frac{d}{b+d} + \frac{d}{c+d}$
Sokal & Sneath Similarity Measure 2	$m_2 = \frac{ad}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$
Sokal & Sneath Similarity Measure 3	$m_3 = \frac{2(a+d)}{2(a+d)+b+c}$
Sokal & Sneath Similarity Measure 4	$m_4 = \frac{a}{a+2(b+c)}$
Sokal & Sneath Similarity Measure 5	$m_5 = \frac{a+d}{b+c}$
Kulczynski Similarity Measure 1	$m_6 = \frac{a}{b+c}$
Ochiai Similarity Measure	$m_7 = \sqrt{\left(\frac{a}{a+b}\right)\left(\frac{a}{a+c}\right)}$
Binary Lance and Williams Nonmetric Dissimilarity Measure	$m_8 = \frac{b+c}{2a+b+c}$
Pattern Difference	$m_9 = \frac{bc}{(a+b+c+d)^2}$
Binary Minimum Histogram Difference	$dm_{10} = \sum_{n=1}^{4} \boldsymbol{min}(p_n^\beta, p_n^{\beta+1})$
Binary Absolute Histogram Difference	$dm_{11} = \sum_{n=1}^{4} \lvert p_n^\beta - p_n^{\beta+1} \rvert$
Binary Mutual Entropy	$dm_{12} = -\sum_{n=1}^{4} p_n^\beta \boldsymbol{log}\, p_n^{\beta+1}$
Binary Kullback Leibler Distance	$dm_{13} = -\sum_{n=1}^{4} p_n^\beta \boldsymbol{log} \frac{p_n^\beta}{p_n^{\beta+1}}$
Ojala Minimum Histogram Difference	$dm_{14} = \sum_{n=1}^{N} \boldsymbol{min}(S_n^\beta, S_n^{\beta+1})$
Ojala Absolute Histogram Difference	$dm_{15} = \sum_{n=1}^{N} \lvert S_n^\beta - S_n^{\beta+1} \rvert$
Ojala Mutual Entropy	$dm_{16} = -\sum_{n=0}^{15} S_n^\beta \boldsymbol{log}\, S_n^{\beta+1}$
Ojala Kullback Leibler Distance	$dm_{17} = -\sum_{n=1}^{N} S_n^\beta \boldsymbol{log} \frac{S_n^\beta}{S_n^{\beta+1}}$

We have used three types of binary similarity measures between bit planes as in Table 1.

First group:. The measures m_1 to m_9 are obtained for neighbor bits separately by applying the parameters moments {*a,b,c,d*} in (3) to the binary string similarity measures, such as Sokal & Sneath.

Second group: The differences $dm_i = m_i^\beta - m_i^{\beta+1}$ $i = 10, \ldots, 13$ are used as the final measures.

Third group: Measures dm_{14}-dm_{17} are the neighborhood-weighting mask proposed by Ojala [4].

3. EXPERIMENTAL RESULTS

We computed binary similarity measures as features and used Sequential Floating Forward Search (SFFS) algorithm to select the best features [5] and we have used Linear Regression Classifier for classification [7]. In our experiments we have built a database of 200 images. These images were taken with Canon Powershot S200 camera. Notice that the images that were taken from the same camera in order to detect alterations, but not the properties due to the camera characteristics.

The image alterations we experimented with were scaling-up, rotation, brightness adjustment, blurring and sharpening, all implemented via Adobe Photoshop [8]. Half of the images were used for training and the remaining in testing. In [2], Farid *et al.* employed a higher order statistical model to discriminate natural images from unnatural ones. We have adopted their method, so that we did the same tests once with their features and then with our features. In the Table's below the results according to features in [2] are denoted as "Farid".

First, we scaled-up all the images with the scales of %50, %25, %10, %5, %2, %1 and got 6 databases of 200 images. We trained a classifier on each database and tested if an image is original or scaled-up. The results are in Table II.

Table II. The performance for image scaling-up attack.

Scaling-up	Method	False Positive	False Negative	Accuracy (%)
%50	BSM	2/100	0/100	99
	Farid	4/100	11/100	92.5
	Farid	5/100	11/100	92
%10	BSM	18/100	3/100	89.5
	Farid	4/100	17/100	89.5
%5	BSM	25/100	4/100	85.5
	Farid	4/100	14/100	91
	Farid	8/100	21/100	85.5
%1	BSM	32/100	8/100	80
	Farid	17/100	12/100	85.5

We rotated the images 45°, 30°, 15°, 5°, 1°. Corresponding results are in Table III.

Table III. The performance for rotation attack.

Rotation	Method	False Positive	False Negative	Accuracy (%)
%50	BSM	2/100	0/100	99
	Farid	4/100	11/100	92.5
%25	BSM	7/100	0/100	96.5
	Farid	5/100	11/100	92
%10	BSM	18/100	3/100	89.5
	Farid	4/100	17/100	89.5
%5	BSM	25/100	4/100	85.5
	Farid	4/100	14/100	91
%2	BSM	27/100	7/100	83
	Farid	8/100	21/100	85.5

We adjusted the brightness of the images with the scales of 40, 25, 15, 5. Corresponding results are in Table IV.

Table IV. The performance for brightness adjustment attack.

Brightness Adjustment	Method	False Positive	False Negative	Accuracy (%)
40	BSM	17/100	27/100	78
	Farid	60/100	28/100	58
25	BSM	13/100	32/100	77.5
	Farid	61/100	26/100	56.5
15	BSM	19/100	28/100	76.5
	Farid	67/100	27/100	53.5
5	BSM	18/100	45/100	68.5
	Farid	59/100	39/100	51

We use Gaussian blur to blur the images with the scales of 1, 0.5, 0.3, 0.1. Corresponding results are represented in Table-V.

Table V. The performance for blurring attack.

Blurring	Method	False Positive	False Negative	Accuracy (%)
1.0	BSM	1/100	0/100	99.5
	Farid	0/100	7/100	96.5
0.5	BSM	2/100	0/100	99
	Farid	81/100	1/100	59
0.3	BSM	46/100	22/100	66
	Farid	49/100	38/100	56.5
0.1	BSM	24/100	62/100	57
	Farid	69/100	31/100	50

We sharpen the images and train a classifier to distinguish the sharpened ones from the original ones. In Table VI, we show the results of the sharpening classifier.

Table VI. The performance for sharpening attack.

	Method	False Positive	False Negative	Accuracy (%)
Sharpening	BSM	4/100	9/100	93.5
	Farid	36/100	19/100	72.5

As shown in the tables we trained more than one classifier for each image alteration type at different settings of attack strength. However, it is not practical to devise a separate classifier for each setting; hence we trained one classifier per alteration type to operate in a range of attack strengths. For example we generate an image pool with 50 images from %25, %10, %5, and % 2 scaled-up. We used half of the images for training and remained for testing. The results for generic classifier for various image alteration types are given in Table VII.

To test an image on only one classifier we made an image pool by adding the same quantity of images that are scaled up with the scales of %50, %25, %10, %5, scaled down %50, %25, %10, %5, rotated 45°, 30°, 15°, 5°, contrast enhanced with the scales of 25,15,5, brightness adjusted with the scales of 15, 25, blurred with the scales of 0.3, 0.5 and sharpened. Again half of the images were used for training and the remaining for testing. We call this classifier as generic-generic classifier. Corresponding results for this classifier is shown in Table VIII.

Table VII. The performance of generic classifiers.

Image Alteration Type	Method	False Positive	False Negative	Accuracy (%)
Scaling Up	BSM	12/100	3/100	92.5
	Farid	6/100	17/100	88.5
Scaling Down	BSM	29/100	13/100	79
	Farid	17/100	18/100	82.5
Rotation	BSM	13/100	45/100	71
	Farid	16/100	14/100	85
Contrast Enhancement	BSM	1/100	48/100	75.5
	Farid	79/100	13/100	54
Brightness Adjustment	BSM	3/100	46/100	75.5
	Farid	76/100	17/100	53.5
Blurring	BSM	6/100	18/100	88
	Farid	80/100	4/100	58

Table VIII. The performance of generic-generic classifiers.

Method	False Positive	False Negative	Accuracy (%)
BSM	21/100	28/100	75.5
Farid	15/100	31/100	77

To make our results more realistic, we addressed the testing of "doctored images". We doctored 20 images by either inserting extra content or replacing the original content. To make them look like natural and avoid any suspicion, the inserted content was resized, rotated or brightness adjusted etc, before pasting it to the image. We take 2 untampered and one tampered block from every image, so we had 40 untampered and 20 tampered blocks. We tested these blocks on generic classifiers. We accept it as tampered if any of the generic classifiers declare it as tampered. In Table IX the results for the image blocks on generic classifiers are shown.

Table IX. The perf. of generic classifiers for image blocks.

Method	False Positive	False Negative	Accuracy (%)
BSM	9/40	2/20	81.67
Farid	40/40	0/20	33.3

And we tested the same blocks on generic - generic classifiers. The corresponding results are in Table X.

Table X. The perf. of generic classifiers for image blocks.

Method	False Positive	False Negative	Accuracy (%)
BSM	8/40	4/20	80
Farid	9/40	8/20	71.67

We capture 100 images from Internet that can easily be tampered. We tested these images on generic and generic - generic classifiers. The results are shown in Table XI and Table XII.

Table XI. The performance of generic classifiers for image blocks that are captured from Internet.

Method	False Negative	Accuracy
BSM	9/100	91
Farid	0/100	100

Table XII. The performance of generic-generic classifiers for image blocks that are captured from internet.

Method	False Negative	Accuracy (%)
BSM	48/100	52
Farid	47/100	53

4. CONCLUSIONS

In this paper we proposed a method for digital image forensics, based on Binary Similarity Measures between bit planes used as features. Then we designed several classifiers to test the tampered or un-tampered status of the images. The performance results in detecting and differentiating a host of attacks were encouraging as we were able to discriminate a doctored image from its original with a reasonable accuracy. We have assessed our methods vis-à-vis the closest competitor image forensic detector in [2]. We outperform Farid's detector especially in contrast enhancement and brightness adjustment attacks. On the other hand, while we have better performance at stronger levels of manipulations, Farid outperforms us at weaker levels. In this respect, the two schemes seem to be complementary; hence fusion of forensic detectors at feature level or decision level must be envisioned.

5. REFERENCES

[1] Avcıbaş, İ., N. Memon, B. Sankur. 2002. Image Stegalysis with Binary Similarity Measures, Proceedings of International Conference on Image Processing, Volume 3, 645-648, 2002.

[2] Farid, H., S. Lyu. Higher-Order Wavelet Statistics and their Application to Digital Forensics, IEEE Workshop on Statistical Analysis in Computer Vision (in conjunction with CVPR), Madison, Wisconsin, 2003.

[3] İ. Avcıbaş, I., M. Kharrazi, N. Memon, B. Sankur, Image Steganalysis with Binary Similarity Measures, Applied Signal Processing (under review)

[4] Ojala, T., M. Pietikainen, D. Harwood. A Comparative Study of Texture Measures with Classification Based on Feature Distributions, Pattern Recognition, vol.29, pp, 51-59.

[5] Pudil, P., F. J. Ferri, J. Novovicov and J. Kittler. Floating search methods for feature selection with nonmonotonic criterion functions. In Proceedings of the 12th ICPR, volume 2, pages 279-283, 1994.

[6] Avcıbaş, İ., S. Bayram, N. Memon, M. Ramkumar, B. Sankur, A Classifier Design for Detecting Image Manipulations, Proceedings of International Conference on Image Processing, 2004, Singapore.

[7] A. C. Rencher, *Methods of Multivariate Analysis*, New York, John Wiley (1995).

[8] www.adobe.com