Tampered Region Localization of Digital Color Images Based on JPEG Compression Noise

Wei Wang, Jing Dong, and Tieniu Tan

National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Sciences,
P.O. Box 2728, Beijing, P.R. China, 100190
{wwang,jdong,tnt}@nlpr.ia.ac.cn

Abstract. With the availability of various digital image edit tools, seeing is no longer believing. In this paper, we focus on tampered region localization for image forensics. We propose an algorithm which can locate tampered region(s) in a lossless compressed tampered image when its unchanged region is output of JPEG decompressor. We find the tampered region and the unchanged region have different responses for JPEG compression. The tampered region has stronger high frequency quantization noise than the unchanged region. We employ PCA to separate different spatial frequencies quantization noises, i.e. low, medium and high frequency quantization noise, and extract high frequency quantization noise for tampered region localization. Post-processing is involved to get final localization result. The experimental results prove the effectiveness of our proposed method.

Keywords: Image forensics, Tampered region localization, JPEG compression noise, PCA.

1 Introduction

Along with the rapid development of image editing software (e.g. Adobe Photoshop), digital images can be easily manipulated and tampered images can hardly be detected by human eyes. Seeing is no longer believing. It is necessary to develop authentication techniques to verify the integrity of a digital image.

Generally speaking, there are two types of approaches for image authentication: active [3, 4] and passive [13, 19] approaches. Active approaches often require pre-processing (e.g. watermark embedding or signature generating), and they are not desired for practical use in daily life since the image capture devices are not usually all integrated with watermarking embedding module. Passive approaches, which gather evidence of tampering from images themselves, however, have more potential for practical use and gains more attention among researches in image forensics.

We focus on passive approaches and try to locate the tampered region in a tampered image. Tampered region(s) localization in tampered image is more meaningful and convincible than simple detection of existence of tampered image for image forensics. Tampered image detection can only tell us whether an

H.-J. Kim, Y. Shi, and M. Barni (Eds.): IWDW 2010, LNCS 6526, pp. 120–133, 2011.

image is tampered or not. However, we do not know whether it is the tampering operation or other operations (e.g. JPEG compression) that affect information for tampered images. Whereas, tampered region localization can directly imply where the tampering operation occurs. In this paper, we will propose an algorithm which can locate tampered region(s) in a lossless compressed tampered image when its unchanged region is output of JPEG decompressor.

For such a tampered image, we find the tampered region and the unchanged region have different responses for JPEG compression. The unchanged region has weaker high frequency quantization noise than the tampered region. We then employ principle component analysis (PCA) to separate different spatial frequencies quantization noises, i.e. low, medium and high frequency quantization noise, and extract high frequency noise for tampered region localization.

The rest of this paper is organized as follows. Some related works are introduced in Section 2. Section 3 mainly introduces our proposed algorithm for tampered region localization. The experimental results and analysis are given in Section 4. Conclusions are drawn in Section 5.

2 Related Works

In recent years, many researchers focus on digital image tampering detection and have proposed a number of techniques. There are several methods for passive image tampering detection proposed in the recent literature [2, 5, 6, 7, 8, 9, 10, 11, 15, 16, 17, 18, 20, 21].

Farid et al. have done pioneering work in this area. In [6] and [7], *Johnson* and *Farid* developed a technique of tampering detection by analyzing the inconsistency of lighting in image. But it may fail when source images used for tampering are taken under similar lighting conditions. Besides, it needs to manually select the points near the boundary of suspicious object. *Popescu* and *Farid* [17] argued that color interpolation (demosaicing) introduced specific correlations between neighboring pixels of a color image, while image tampering might destroy or alter them and based on this they proposed an image tampering detection algorithm to check the periodicity of these correlations. Actually, they did not try their method on real tampered examples. Besides, in [2], *Dirik* and *Memon* utilized artifacts created by Color Filter Array (CFA) to detect image tampering. They proposed two features for tampering detection. One is based on CFA pattern estimation and the other is based 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. Actually, CFA artifacts are hardly detected for many images with heavy JPEG compression. In [11] and [16], authors assumed that image tampering would involve resampling. They proposed approaches to detect periodicity of correlations introduced by resampling. However, they did not give enough real examples for tampered region localization. *Lukáš et al.* [10] proposed a digital image tampering detection method to detect camera pattern noise which is considered as a unique stochastic characteristic of imaging sensor. The tampered region is determined when image region is

detected as lacking of the pattern noise. However, this method is only applicable when the tampered image is claimed to have been taken by a known camera or at least we have images taken by the camera before. In [8], *Krawetz* proposed a suit of tools to analyze images and do forensics. He did a series of experiments rather than deep analysis. *Shi et al.* [18] proposed a splicing detection method using effective features extracted from image Markov transfer matrices. Experiments were carried on Columbia image splicing detection evaluation dataset [14] and the results were satisfying. Aiming at color image tampering detection, we proposed an effective color image tampering detection approach based on image chroma [20, 21]. We found that the analysis on chroma of color image was more reasonable for tampered image detection than on illuminance because chroma could reflect more information left by tampering which human eyes might not observe. If we use the proposed methods in [10, 11, 16, 17, 18, 20, 21] to find the tampered region by sliding window within an image, we should carefully choose the window size. Too small will not have enough statistical information while too big will not locate accurately.

There are also some methods for JPEG image forensics since JPEG is the most widely used image format. Double JPEG compression can be a cue for image tampering, but detecting double JPEG compression [12, 15] does not necessarily prove malicious tampering. *He et al.* [5, 9] proposed a workable method by using the double JPEG quantization effect hidden among the DCT coefficients to automatically detect the tampered regions of images. They agreed that the unchanged region in a tampered JPEG image undergoing double JPEG compression while the tampered region undergoing only once. They tried to use the inconsistency to locate the tampered region. However, the average detection rate both in image level and region level are below 65%; and their method are sensitive to the estimation of the period.

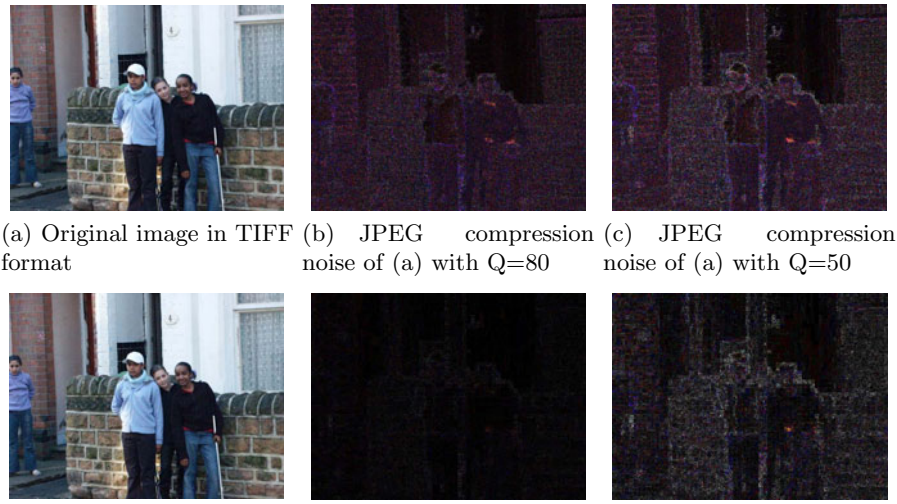
3 Proposed Approach

Basically, all image manipulations can be roughly classified into local changing and global changing operations. In this paper, we focus on local changing operation. We want to locate the local changing, i.e. tampered region. We define a tampered image as in [5, 9]. *Lin et al.* regards an image as tampered one when part of its content has been altered. In other words, that an image is tampered implies that it must contain two parts: the unchanged region and the tampered region.

Since JPEG is the most widely used image format, we mainly focus on locating tampered regions in a lossless compressed image when the unchanged region of the image is output of JPEG decompressor. We will utilize properties of JPEG compression to locate the tampered region.

3.1 JPEG Compression Noise

JPEG compression noise can be simply calculated by subtracting a given image from its JPEG compressed version. Different responses can be get if we



(a) Original image in TIFF format (b) JPEG compression noise of (a) with Q=80 (c) JPEG compression noise of (a) with Q=50

(d) JPEG format of (a) with Q=75 (e) JPEG compression noise of (d) with Q=80 (f) JPEG compression noise of (d) with Q=50

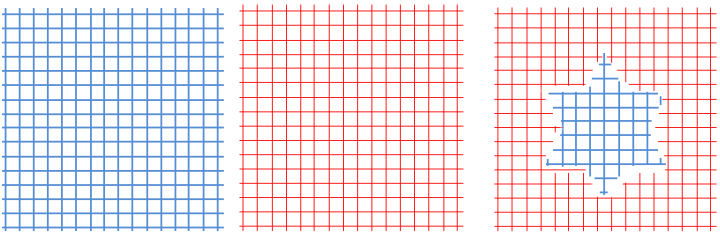
Fig. 1. JPEG compression noises

compress image originally stored in lossless compressed format and the same one in JPEG compressed format respectively. Fig. 1 shows noise images of JPEG compressions of TIFF image and its JPEG version. All the JPEG compressions use the standard JPEG quantization tables recommended by Independent JPEG Group (IJG). From Fig. 1, we find that (b) and (c) have different kinds of noise from (e) and (f). Hence, we can draw a conclusion that with the same quality Q, JPEG compression noise of an original lossless compressed image is quite different from that of its JPEG compressed version no matter the compression quality of the JPEG compressed version is smaller or bigger than Q. However, we prefer bigger one since there are much more difference between Fig.1(b) and (e) than that between (c) and (f).

As long as the unchanged region of a tampered image has been compressed by JPEG previously, the JPEG compression noise of unchanged region is different from that of the tampered region if we compress the whole tampered image with high quality. There are several reasons [9]:

1. If the tampered region comes from the a BMP image or other lossless compressed format image, the tampered region will have different noise as we see in Fig. 1.
2. If the tampered region comes from other JPEG image and its JPEG grid[1] is mismatched with that of the unchanged region, we can consider it as without undergoing JPEG compression before (see Fig. 2). Fig. 2 illustrates

[1] A JPEG grid is the horizontal lines and the vertical lines that partition an image into 8×8 blocks during JPEG compression.



(a) JPEG grids of two JPEG image (b) The tampered image of (a)

Fig. 2. Illustration of tampering two JPEG image. The JPEG grids of the blue region and the red region in (b) are mismatched.

the tampering operation of two JPEG images. If we compress Fig. 2(b), the tampered region (blue grid) can be considered as only undergoing once JPEG compression while the unchanged region (red grid) undergoing twice. Besides, the tampered region may undergo pre-processing (e.g. resizing or (and) rotation) which makes it like never JPEG compressed before. Hence, the JPEG compression noise should be different between these two regions.

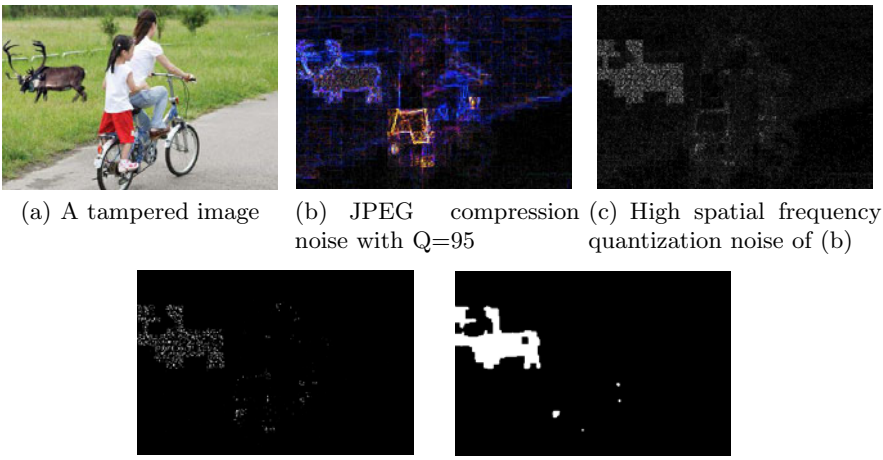
3. Even if the JPEG grids of the tampered region and the unchanged region are matched, the 8×8 blocks along the boundary of the tampered region will consist of pixels in the tampered region and also pixels in the unchanged region. These blocks have different noise from others.

When a tampered image is compressed, the unchanged region actually undergoes double JPEG compression and the tampered region can be considered as being compressed only once. If the compression is with high quality (compressed slightly), for the unchanged region, most high frequencies are erased by previous JPEG compression, hence, for the second JPEG compression, the noise almost comes from quantization of low and medium frequencies. High frequency DCT coefficients are already quantized to zeros by previous JPEG compression. However, for the tampered region, which only undergoes once JPEG compression, its compression noise contains low, medium and high frequency quantization noise. Therefore, the JPEG compression noise of the tampered image consists of two different regions. This is the basic idea of our approach for locating tampered region. It motivates us to use JPEG compression to compress a suspicious image and check whether its noise contains two different regions, just like in Fig. 3. The tampered image in Fig. 3 is generated by two different JPEG images. The animal in the right-bottom of the image is copied from another JPEG image. From Fig. 3 we can see that the unchanged region and the tampered region have different noises for JPEG compression with $Q = 95$.

The idea in this section is enlightened by [8]. *Krawetz* calls this phenomenon error level analysis. He intentionally resaves a given image at a known JPEG compression quality and calculates the difference between these two images. The tampered region will be found by just watching the difference.

(a) A tampered image (b) JPEG compression noise with Q=95

Fig. 3. A real example of a tampered image. The animal in the right-bottom of the image is the tampered region. The tampered image is generated by two different JPEG images.



(a) A tampered image (b) JPEG compression noise with Q=95 (c) High spatial frequency quantization noise of (b)

(d) Normalized highspatial frequency quantization noise (e) The mask of the tampered region location of (a)

Fig. 4. Tampered region localization of a tampered image. The animal in the middle left of the image is the tampered region.

3.2 Principal Component Analysis

Actually, only using the above idea is not enough for locating the tampered region. We are hardly able to tell the tampered region from the unchanged one sometimes just by human visual perception of JPEG compression noise. Fig. 4 shows an example of tampered image and its JPEG compression noise with quality of 95. The animal in the middle left of the image is the tampered region. However, someone may think the red shorts of the girl is tampered after seeing Fig 4(b). We need deeper analysis. As we mentioned above, JPEG compression noise is related to the quantization step. It can be roughly divided into three components: low, medium and high spatial frequency quantization noise. Low spatial frequency quantization noise comes from quantizing low frequencies DCT

coefficients while high frequency noise comes from quantizing high frequencies DCT coefficients. The biggest difference between the noises of two regions of a tampered image is high spatial frequency quantization noise. However, the noises in the above figures appear in RGB color space. In other words, they are composed of red, green and blue spectrum compression noises. Hence, how to extract high spatial frequency quantization noise from the apparent spectrum noise of JPEG compression should be a key point of tampered region localization.

For each pixel of an image, each component value (R, G, B) can be expressed as weighted combination of 8×8 DCT coefficients, as shown in equation (1). Each pixel can be considered as contains 64 spatial frequencies information. Hence, we can extract spatial frequency information from RGB values.

$$f(x,y) = \sum_{\mu=0}^{8}\sum_{\nu=0}^{8} \alpha(\mu)\alpha(\nu)C(\mu,\nu)\cos[\frac{\pi(2x+1)\mu}{16}]\cos[\frac{\pi(2y+1)\nu}{16}], \quad (1)$$

where $f(x,y)$ is component value at location (x,y) in spatial domain and $C(\mu,\nu)$ is DCT coefficient. $\alpha(\mu)$ is defined as

$$\alpha(\mu) = \begin{cases} \sqrt{1/8} & \text{for } \mu = 0 \\ \sqrt{2/8} & \text{for } \mu \neq 0 \end{cases},$$

It is well known that DCT is used in JPEG compression since it can decorrelate image data to achieve better compression. Different frequencies of DCT coefficients are nearly uncorrelated which can be justify by Fig 5. Fig 5 shows correlation coefficients distribution of 8×8 block DCT frequencies coefficients of an authentic image. We can find most of correlation coefficients are below 0.15 which means different frequencies coefficients are uncorrelated.

Therefore, different spatial frequencies quantization noise of JPEG compression should be uncorrelated. We employ PCA to extract them from RGB values, because PCA involves a mathematical procedure that transforms possibly correlated variables into uncorrelated variables (components). The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability

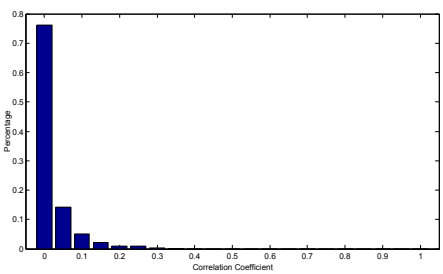


Fig. 5. Correlation coefficients distribution of 8×8 block DCT frequencies coefficients of an authentic image

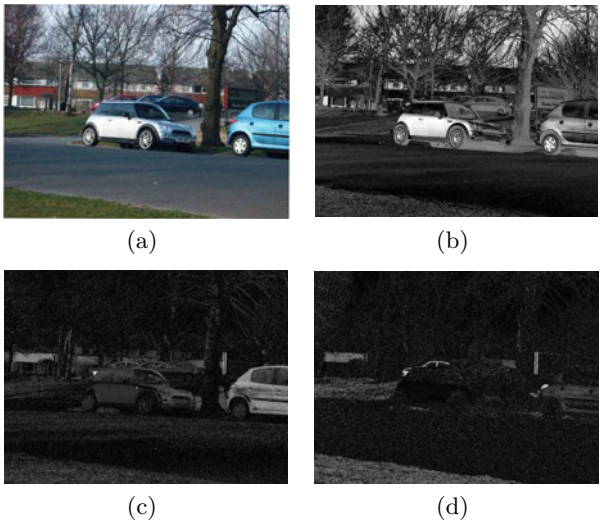


Fig. 6. PCA of an uncompressed image. (a)an uncompressed color image. (b) first PCA component of (a). (c) second PCA component of (a). (d) third PCA component of (a).

as possible [22]. PCA is theoretically the optimum transform for given data in least square terms. Fig 6 shows an example of PCA of an uncompressed color image, from which we can find that the third component of PCA is actually high frequency information (noise).

Hence, for JPEG compression noise of a given image, we take each pixel as an observation and its RGB value, i.e. red, green and blue spectrum compression noise as variables. In this way, we can get the original data set X. X is an $N \times 3$ matrix where N is the number of pixels of the given image. Our goal is to get another $N \times 3$ matrix by a linear transformation P, as shown in equation (2), so that components of re-expressed data are de-correlated.

$$Y = XP \,. \tag{2}$$

Of the re-expressed JPEG compression noise, high spatial frequency quantization noise should be the smallest variance component. As stated above, JPEG compression noise of the tampered region has stronger high spatial frequency quantization noise than that of the unchanged region. Hence, we extract high spatial frequency quantization noise to locate the tampered region.

3.3 Post-Processing

In this section, we introduce post-processing operations on the high frequency quantization noise to try to locate the tampered region. The high frequency quantization noise of a tampered image should have obviously two parts: concentrated high values region (probably the tampered region) and low values

region (probably the unchanged region). In its high values region, there may be low noise values scattered because not everywhere in the tampered region is high frequency information. We want to find these high noise values and use morphology operation to locate the tampered region.

We first employ sigmoid function (3) to normalize the high frequency noise value t to $P(t)$ within range of $[0, 1]$.

$$P(t) = \frac{1}{1 + e^{-a(t-b)}}, \tag{3}$$

where a controls the shape of function and b is determined by the high frequency noise. Fig. 4(d) shows normalized high frequency quantization noise for which $a = 3$ and b equals mean of high frequency noise plus three times of its variance.

Beyond normalizing, we also explore some morphology operations since the high value noise is not very close to each other but they are concentrated. Fig 4(e) shows the tampered region locating result of Fig 4(a).

3.4 Algorithm Overview

To summarize, our proposed algorithm for tampered region localization of digital color image are shown as follows:

Algorithm 1. Our tampered region localization algorithm

Input:
a suspicious image I

Output:
a mask of the tampered region localization result M;

1: Resave image I to JPEG image I' with quality Q;
2: $Noise = I - I'$;
3: $[repreNoise] = PCA(Noise)$;
4: Extract high frequency noise $High\_Noise = repreNoise(:, :, 3)$;
5: Normalize the high frequency noise using sigmoid function. $High\_Noise\_Norm = sigmoid(High\_Noise, a, b)$;
6: Post-processing normalized high frequency noise using morphology operations $M = imopen(imclose(High\_Noise\_Norm))$;
7: return M;

4 Experiments

We used our public color image dataset CASIA TIDE v2.0 [1] in our experiments. It consists of 7,491 authentic and 5,123 sophisticatedly tampered color images of different sizes, varying from 240×160 to 900×600. We randomly chose some tampered image in TIFF format to show here to check the effectiveness of our proposed approach. In our experiments, we set JPEG compression quality to 95 to get compression noise. We also let $a = 3$ and b equals mean of high frequency noise plus three times of its variance for sigmoid function (3). For

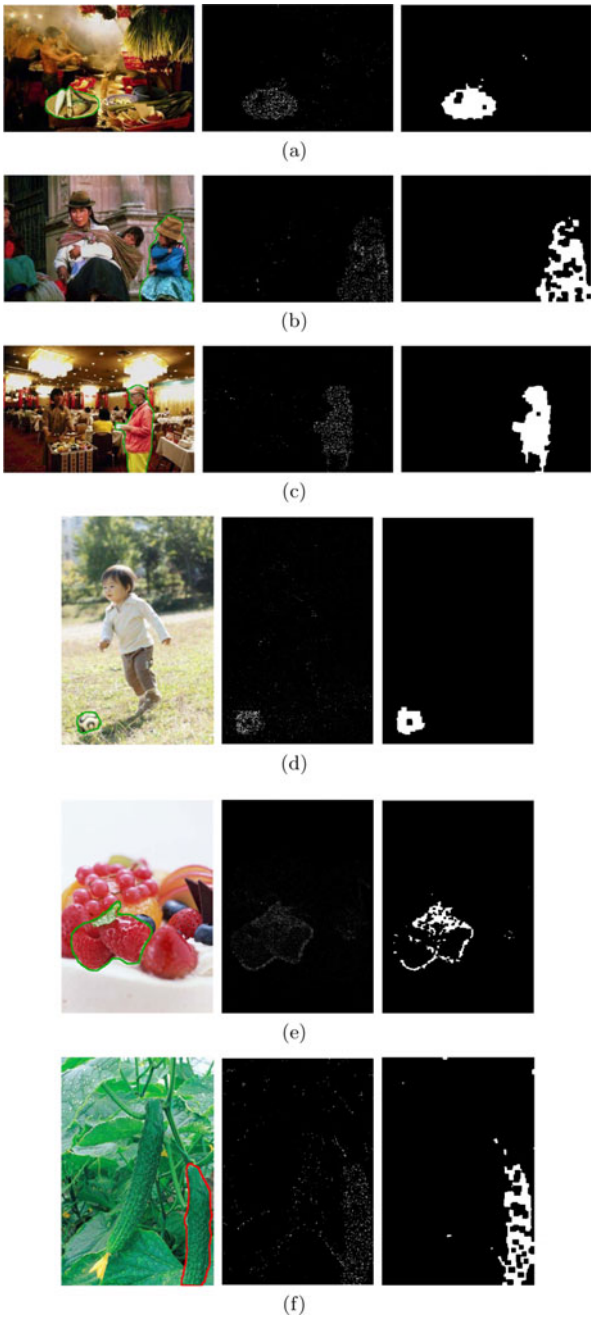


Fig. 7. Experimental results of tampered images. First column shows tampered images with ground truth marked by red or green contour. Second column is the normalized highspatial frequency quantization noises. Last column shows the masks of the tampered region locations given by our proposed algorithm.

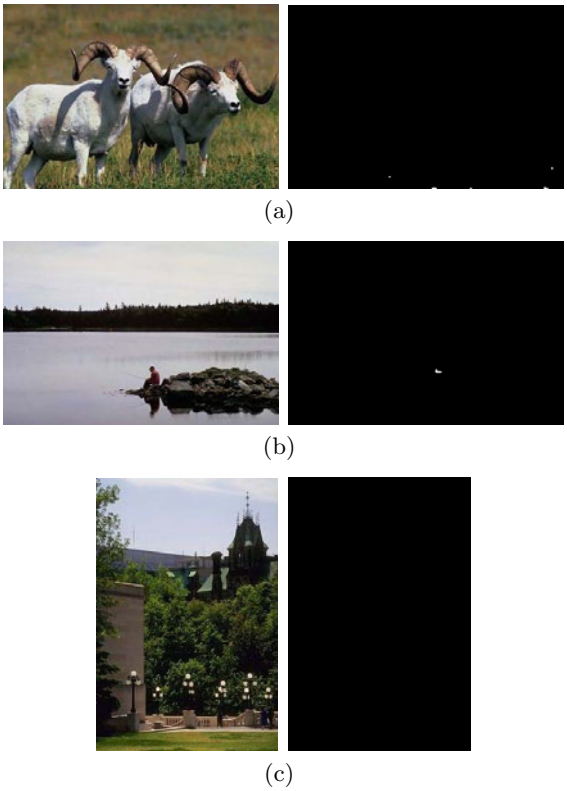


Fig. 8. Experimental results of authentic images. First column shows authentic images. Second column shows the masks of the tampered region locations given by our proposed algorithm. No tampered regions were found except some false alarm points.

morphology operations, we used matlab morphology function *imclose* with 8×8 square structure which followed by *imopen* operation with 3×3 square structure to get locations of tampered regions. All these parameters were set empirically. Fig. 7 shows the localization results of some tampered images in CASIA TIDE v2.0. First column shows tampered images with ground truth marked by red or green contour. Second column is their normalized highspatial frequency quantization noises. Last column shows the masks of the tampered region locations given by our proposed algorithm. Fig. 8 shows the localization results of some authentic images in CASIA TIDE v2.0. First column shows authentic images. Second column shows localizations results with no tampered regions being found except some false alarm points. Since not all tampered regions have enough high frequency information, the localization results of the tampered region, i.e. the white regions in masks are not always connected.

Fig. 9 shows some unsuccessful cases in our experiments, in which (a) and (b) are tampered images and its localization results, while (c) is an authentic

(a) A tampered image with background (sky) being substituted and its localization result.

(b) A tampered image of adding a flower bud in right bottom of the image and its localization result in middle column. Last column shows its localization results with JPEG compression quality $Q = 100$.

(c) An authentic image and its localization result in middle column. Last column shows its localization results with JPEG compression quality $Q = 100$.

Fig. 9. Some unsuccessful cases. (b) is a tampered image based on (c). (c) is original saved in JPEG format with quality Q=100.

image with its localization results. The image in (a) is a tampered image with background (sky) being substituted. Since the sky is almost low frequency information, we cannot use its high frequency JPEG compression noise to locate it. However, our algorithm successfully located the boundary of the tampered region. The boundary consists of pixels from the unchanged region and the tampered region. It can be considered as never JPEG compressed region. That is why our algorithm can locate it. Nonetheless, we cannot tell which part of the image is tampered from the located boundary. Hence, when the tampered region has little high frequency information, our method may fail. The image in (b) is a tampered image of (c) by adding a flower bud in the bottom right of it. Since the unchanged region of the image in (b), i.e. parts of the image in (c) is JPEG compressed with quality $Q = 100$ which can be considered as with no lossy compression, if we use $Q = 95$ to compress the image to get its JPEG compression noise, both the unchanged region and the tampered region will have strong high frequency noise. The localization result will not be correct, like middle column shows in (b). When we use $Q = 100$ to compress the image, the localization

result (last column in (b)) are correct. For the authentic image in (c), we will get false tampered region(s) localization result by compressing it with $Q = 95$, while using $Q = 100$ we can get the correct localization result. Fig. 9 (b) and (c) failed because of our underlying assumption that all JPEG format images used for tampering are saved with a quality below a reasonable value (in our experiments, we assume this value is 95). We will focus more on the estimation of Q factor in our future work.

5 Conclusions

In this paper, we have proposed an algorithm which can locate the tampered region in a lossless compressed tampered image when its unchanged region is output of JPEG decompressor. We have utilized different responses for JPEG compression of the tampered region and the unchanged region as the cue for tampered region localization. The tampered region always has some high frequency information while that of the unchanged region is almost erased by previously JPEG compression. The experimental results have proved the effectiveness of our proposed algorithm. However, if the tampered region of a tampered image has little high frequency information or the source image of its the unchanged region saved in JPEG format with higher quality than the quality we used in our experiments, our algorithm may fail. The unsuccessful cases in later situation alert us that we should estimate the JPEG compression history of a given image first and then use the reasonable quantization matrices to compress the image to do further analysis in our future work. Beside, making use of double JPEG effect like *He et al.* [5, 9] proposed approach to improve our proposed approach should also be considered in our future work.

Acknowledgments. This work is funded by National Basic Research Program (Grant No. 2004CB318100), National Natural Science Foundation of China (Grant No. 60736018, 60702024, 60723005), and National Hi-Tech R&D Program (Grant No. 2006AA01Z193, 2007AA01Z162).

References

1. CASIA Tampered Image Detection Evaluation Database (2010), `http://forensics.idealtest.org`
2. Dirik, A., Memon, N.: Image tamper detection based on demosaicing artifacts. In: IEEE International Conference onImage Processing (ICIP), pp. 1497–1500 (2009)
3. Feng, W., Liu, Z.-Q.: Region-level image authentication using bayesian structural content abstraction. IEEE Transactions on Image Processing 17(12), 2413–2424 (2008)
4. Haouzia, A., Noumeir, R.: Methods for image authentication: a survey. Multimedia Tools and Applications 39(1), 1–46 (2008)
5. He, J., Lin, Z., Wang, L., Tang, X.: Detecting doctored JPEG images via DCT coefficient analysis. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 423–435. Springer, Heidelberg (2006)

6. Johnson, M.K., Farid, H.: Exposing digital forgeries by detecting inconsistencies in lighting. In: ACM Multimedia and Security Workshop, pp. 1–10 (2005)
7. Johnson, M.K., Farid, H.: Exposing digital forgeries in complex lighting environments. IEEE Transactions on Information Forensics and Security 2(3), 450–461 (2007)
8. Krawetz, N.: A picture's worth: Digital image analysis and forensics (August 2007), http://www.hackerfactor.com/
9. Lin, Z., He, J., Tang, X., Tang, C.K.: Fast, automatic and fine-grained tampered jpeg image detection via dct coefficient analysis. Pattern Recognition 42(11), 2492 (2009)
10. Lukáš, J., Fridrich, J., Goljan, M.: Detecting digital image forgeries using sensor pattern noise. In: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 6072, pp. 362–372 (February 2006)
11. Mahdian, B., Saic, S.: Blind authentication using periodic properties of interpolation. IEEE Transactions on Information Forensics and Security 3(3), 529–538 (2008)
12. Mahdian, B., Saic, S.: Detecting double compressed jpeg images. In: IET Seminar Digests 2009, vol. (2), p. P12 (2009)
13. Ng, T., Chang, S., Lin, C., Sun, Q.: Passive-blind image forensics. In: Multimedia Security Technologies for Digital Rights, ch. 6. Elsevier, Amsterdam (2006)
14. Ng, T., Chang, S., Sun, Q.: A data set of authentic and spliced image blocks. Tech. rep., DVMM, Columbia University (2004), http://www.ee.columbia.edu/ln/dvmm/downloads/AuthSplicedDataSet/photographers.htm
15. Popescu, A.C., Farid, H.: Statistical tools for digital forensics. In: Fridrich, J. (ed.) IH 2004. LNCS, vol. 3200, pp. 128–147. Springer, Heidelberg (2004)
16. Popescu, A.C., Farid, H.: Exposing digital forgeries by detecting traces of resampling. IEEE Transactions on Signal Processing 53(2), 758–767 (2005)
17. Popescu, A.C., Farid, H.: Exposing digital forgeries in color filter array interpolated images. IEEE Transactions on Signal Processing 53(10), 3948–3959 (2005)
18. Shi, Y., Chen, C., Xuan, G.: Steganalysis versus splicing detection. In: Shi, Y.Q., Kim, H.-J., Katzenbeisser, S. (eds.) IWDW 2007. LNCS, vol. 5041, pp. 158–172. Springer, Heidelberg (2008)
19. Wang, W., Dong, J., Tan, T.: A survey of passive image tampering detection. In: Ho, A.T.S., Shi, Y.Q., Kim, H.J., Barni, M. (eds.) IWDW 2009. LNCS, vol. 5703, pp. 308–322. Springer, Heidelberg (2009)
20. Wang, W., Dong, J., Tan, T.: Effective image splicing detection based on image chroma. In: IEEE International Conference on Image Processing, pp. 1257–1260 (2009) (accepted)
21. Wang, W., Dong, J., Tan, T.: Image tampering detection based on stationary distribution of markov chain. In: IEEE International Conference on Image Processing (2010) (accepted)
22. Wikipedia: Principal component analysis — wikipedia, the free encyclopedia (2010), http://en.wikipedia.org/w/index.php?title=Principal_component_analysis\&oldid=366194078 (accessed June 12, 2010)