A Comparison Study on Copy-Cover Image Forgery Detection

Frank Y. Shih* and Yuan Yuan

Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102, USA

Abstract: Due to rapid advances and availabilities of powerful image processing software, digital images are easy to manipulate and modify for ordinary people. This makes it more and more difficult for a viewer to check the authenticity of a given digital image. For digital photographs to be used as evidence in law issues or to be circulated in mass media, it is inevitably needed to identify whether an image is authentic or not. In this paper, we discuss the techniques of copy-cover image forgery and compare four detection methods for copy-cover forgery detection, which are based on PCA, DCT, spatial domain, and statistical domain. We investigate their effectiveness and sensitivity under the influences of Gaussian blurring and lossy JPEG compressions. It is concluded that the PCA method outperforms the others in terms of time complexity and accuracy. In JPEG compression simulation, its true positive rate is above 90% and false positive rate is above 99%. In Gaussian blurring simulation, its true positive rate is above 77% and false positive rate is above 99%.

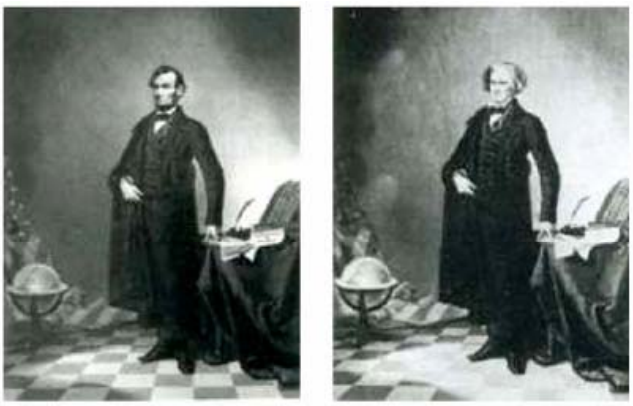
Keywords: Digital forensics, copy-cover detection, image forgery, and image splicing.

1. INTRODUCTION

It was a very difficult task in the old time without digital cameras and computers to create a good splicing photograph, which requires sophisticated skill of darkroom masking. Due to rapid advances in powerful image processing software, digital images are easy to manipulate and modify. This makes it more difficult for humans to check the authenticity of a digital image. Nowadays, modifying the content of digital images becomes much easier with the help of sophisticated software such as Adobe Photoshop™. It was reported that there are five million registered users of Adobe Photoshop™ up to year 2004 [1]. Image editing software is generally available, and some of them are even free, such as GIMP™ (the GNU Image Manipulation Program) and Paint.Net™ (the free image editing and photo manipulation software designed to be used in computers that run Microsoft Windows™). The ease of creating faked digital images with a realistic quality makes us think twice before accepting an image as authentic. For the news photographs and the electronic check clearing systems, image authenticity becomes extremely critical.

As an example of image forgery after the U.S. Civil War, a photograph of Lincoln's head was superimposed onto a portrait of the southern leader John Calhoun, as shown in Fig. (**1**). Another example of image forgery appeared in a video of Osama bin Laden issued on Friday, September 7, 2007 before the sixth anniversary of 9/11. According to Neal Krawetz of Hactor Factor, an expert on digital image forensics, this video contained many visual and audio splices, and all of the modifications were of very low quality.

Checking the internal consistencies within an image, such as whether the shadow is consistent with the lighting or the objects in an image are in a correct perspective view, is one method to examine the authenticity of images. Minor details of faked images are likely to be overlooked by forgers, and thus it can be used to locate possible inconsistency. However, minor or ambiguous inconsistencies can be easily argued unless there are major and obvious inconsistencies. Moreover, it is not difficult for a professional to create a digital photomontage without major inconsistencies.



(a) (b)

Fig. (1). The 1860 portrait of (**a**) President Abraham Lincoln and (**b**) Southern politician John Calhoun (Courtesy of Hoax Photo Gallery).

In this paper, we describe and compare the techniques of copy-cover image forgery detection. It is organized as follows. Section 2 reviews watermarking technique for image authentication. Section 3 presents four copy-cover detection methods, including Principal Component Analysis (PCA), Discrete Cosine Transform (DCT), spatial domain, and statistical domain. Section 4 compares the four copy-cover detection methods, and provides the effectiveness and sensitivity under variant additive noises and lossy Joint Photographic Experts Group (JPEG) compressions. Finally, we draw conclusions in Section 5.

*Address correspondence to this author at the Department of Computer Science, New Jersey Institute of Technology, Newark, NJ 07102, USA; Tel: 973-596-5654; Fax: 973-596-5777; E-mail: frank.y.shih@njit.edu

1874-0618/10

2. WATERMARKING FOR IMAGE AUTHENTICATION

Watermarking is not a brand new phenomenon. For nearly one thousand years, watermarks on papers have been often used to visibly indicate a particular publisher and to discourage counterfeiting in currency. A watermark is a design impressed on a piece of paper during production and used for copyright identification. The design may be a pattern, a logo or an image. In the modern era, as most of the data and information are stored and communicated in a digital form, proving authenticity plays an increasingly important role. As a result, digital watermarking is a process whereby arbitrary information is encoded into an image in such a way that the additional payload is imperceptible to image observers. Fig. (**2**) shows the general procedure of watermarking.

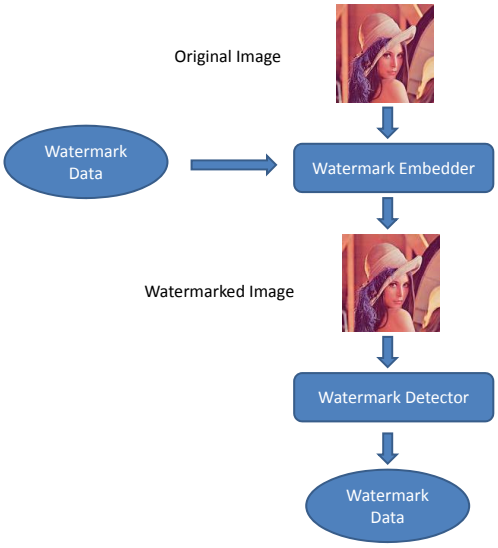


Fig. (2). The general procedure of watermarking.

Digital watermarking has been proposed as a tool to identify the source, creator, owner, distributor, or authorized consumer of a document or an image. It can also be used to detect a document or an image that has been illegally distributed or modified. In a digital world, a watermark is a pattern of bits inserted into a digital media that can identify the creator or authorized users. The digital watermark, unlike the printed visible stamp watermark, is designed to be invisible to viewers. The bits embedded into an image are scattered all around to avoid identification or modification. Therefore, a digital watermark must be robust enough to survive the detection, compression, and operations that are applied on.

In general, watermarking techniques, such as fragile watermarking [2], semi-fragile watermarking [3], or content-based watermarking [4], are often used for the image authentication application. However, watermarking techniques have some drawbacks. Fragile watermark is not suitable for such applications involving compression of images, which is a common practice before sharing images on the Internet. Even though the compression operation is content preserving, fragile watermarking techniques would probably declare a compressed image as unauthentic. Although semi-fragile watermark can be designed to tolerate a specific set of content-preserving operations such as JPEG compression [5], designing such a watermark that can meet the complex requirements of real-life applications is very challenging. It is indeed not easy to develop a watermarking algorithm that can resist such errors produced from scanning and transmission, as well as can tolerate the intensity and size adjustments.

Recently, several watermarking methods have been proposed. Yuan *et al.* [6] and Huang *et al.* [7] put forward integer wavelet based multiple logo-watermarking schemes, in which a visual meaningful logo is embedded in wavelet domain. The watermark is inserted in different bands of wavelet coefficients to make it robust to different attacks. Wu and Cheung [8] presented a reversible watermarking algorithm which exploits the redundancy in the watermarked object to save the recovery information. Kalantari *et al.* [9] proposed a robust watermarking algorithm in the ridgelet domain by modifying the amplitude of the ridgelet coefficients to resist additive white noise and JPEG compression. Luo *et al.* [10] developed a watermarking algorithm using interpolation techniques to restore the original image without any distortion after the hidden data is extracted. Kang *et al.* [11] proposed a watermarking algorithm which is resilient to both print-scan process and geometric distortion by adopting a log-polar mapping.

Since the watermark generation and embedding techniques are closely coordinated in the process of watermarking, the overall success of detection relies upon the security of the watermark generation and embedding. There are several issues to be considered: (1) how easy it is to disable the embedding of watermark, (2) how easy it is to hack the embedding procedure, and (3) how easy it is to generate a valid watermark or embed a valid watermark into a manipulated image. Unfortunately, the embedded watermark can be removed by exploiting the weak points of a watermarking scheme. When a sufficient number of images with the same secret watermark key are obtained, a watermarking scheme can be hacked. There is still not a fully secure watermarking scheme available up-to-date.

3. COPY-COVER IMAGE FORGERY DETECTION

The copy-cover technique is the most popular technique for making image forgery. Copy-cover means that one portion of a given image is copied and then used to cover some object in the given image. If the processing is properly done, most people would not notice that there are identical (or nearly identical) regions in an image. Fig. (**3**) shows an example of copy-cover image forgery, where a region of wall background in the left image is copied and then used to cover two boxes on the wall.

Fig. (3). The image on the left is original, and the image on the right is forged, in which a region of wall background is copied and then used to cover two boxes on the wall.

Several researchers have explored the copy-cover image forgery detection. Mahdian and Saic [12] proposed a method to automatically detect and localize duplicated regions in digital images. Their method is based on blur moment invariants, allowing the successful detection of copy-cover forgery even when blur degradation, additional noise, or arbitrary contrast changes are present in the duplicated regions. These modifications are commonly-used techniques to conceal traces of copy-cover forgery.

Fridrich *et al.* [13] presented an effective copy-cover forgery detection algorithm using DCT and quantization techniques. Popescu and Farid [14] used the PCA domain representation to detect the forged part, even when the copied area is corrupted by noise. Ju *et al.* [15] adopted PCA for small image blocks with fixed sizes, and generates the degraded dimensions representation for copy-cover detection.

Although both DCT and PCA representation methods are claimed to be successful in copy-cover forgery detection, there is a lack of performance comparisons. We evaluate the two methods in terms of time complexity, efficiency and robustness, as well as evaluate two other methods - one is based on spatial domain representation and the other on statistical domain representation.

Given an image of N pixels, our goal is to identify the location and shape of duplicated regions resulting from copy-cover forgery. The general copy-cover detection method is described below. First, a given image is split into small overlapping blocks; each block is transformed into another domain, such as DCT or PCA domain. A two-dimensional matrix is constructed in the way that the pixels in a block are placed in a row by a raster scan and the total number of rows corresponds to the total number of blocks in the given image. By lexicographically sorting all the rows, identical or nearly identical blocks can be detected since they are adjacent to each other. The computational cost of this method is the lexicographic sorting (with time complexity $O(N \log N)$) and domain transformation.

We use four different domain representations for copy-cover detection. The first method is based on PCA domain. The dimension of each block is reduced, and only a number of principal coefficients are preserved. The second method is based on DCT domain. Only a number of most significant DCT coefficients are preserved. Both PCA and DCT methods are in general robust to noise introduced in the process of forgery and can reduce the time consumption in the lexicographical sorting.

The third method is based on spatial domain. All the small blocks are sorted directly according to their pixel values. It saves time since no transformation is involved. However, the lexicographical sorting consumes much more time. The fourth method is based on statistical domain, which uses the mean value and standard deviation of each block for sorting. These four methods are described in more details below.

Before describing the four methods, we need to define the parameter notations. Let N be the total number of pixels in a square grayscale or color image (i.e., the image has $\sqrt{N} \times \sqrt{N}$ pixels in dimensions). Let b denote the number of pixels in a square block (i.e., the block has $\sqrt{b} \times \sqrt{b}$ pixels in dimensions); there are totally $N_b = \left(\sqrt{N} - \sqrt{b} + 1\right)^2$ blocks. Let Nc be the number of principal components preserved in the PCA domain, and let Nt be the number of significant DCT coefficients preserved in the DCT domain. Let Nn denote the number of neighboring rows to search for in the lexicographically sorted matrix. Let Nd be the minimum offset threshold.

3.1. PCA Domain Method

Principal components analysis (PCA) is known as the best data representation in the least-square sense for classical recognition [16]. It is commonly used to reduce the dimensionality of images and retain most information. The idea is to find the orthogonal basis vectors or the eigenvectors of the covariance matrix of a set of images, with each image being treated as a single point in a high dimensional space. Since each image contributes to each of the eigenvectors, the eigenvectors resemble ghostlike images when displayed. The PCA domain method for copy-cover detection is described below.

1. Using PCA, we can compute the new $N_b \times Nc$ matrix representation, in which each row is composed of the first Nc principal components in each block. If a color image is used, we convert the color image into a grayscale image or analyze each color channel separately.
2. Sort the rows of the above matrix in a lexicographic order to yield a matrix S. Let $\vec{s}_i$ denote the rows of S, and let (x_i, y_i) denote the position of the block's image coordinates (top-left corner) that corresponds to $\vec{s}_i$.
3. For every pair of rows $\vec{s}_i$ and $\vec{s}_j$ from S such that $|i - j| < Nn$, place the pair of coordinates (x_i, y_i) and (x_j, y_j) in a list.
4. For all elements in this list, compute their offset, as defined by: $(x_i - x_j, y_i - y_j)$.
5. Discard all the pairs whose offset magnitude, $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$, is less than Nd.
6. Find out the pairs of coordinates with highest offset frequency.
7. From the remaining pairs of blocks, build a duplication map by constructing a zero image, whose size is the same as the original, and coloring all pixels in a duplicated region by a unique grayscale intensity value.

3.2. DCT Domain Method

Many video and image compression algorithms apply the discrete cosine transform (DCT) to transform an image to the frequency domain and perform quantization for data compression [17]. One of its advantages is the energy compaction property; that is, the signal energy is concentrated on a few components while most other components are zero or negligibly small. This helps separate an image into parts (or spectral sub-bands) of hierarchical importance (with respect

to the image's visual quality). The popular JPEG compression technology uses the DCT to compress an image.

We replace step 1 of the aforementioned PCA domain method by DCT to compute the new $N_b \times Nt$ matrix representation, where each row of the matrix is composed of Nt significant coefficients by zigzag ordering of DCT coefficients in each block.

3.3. Spatial Domain Method

We replace step 1 of the PCA domain method by the $N_b \times b$ matrix representation, where each row is the column-wise concatenation of the b pixels in each block.

3.4. Statistical Domain Method

We replace step 1 of the PCA domain method by a $N_b \times 2$ matrix, where each row contains the mean value and the standard deviation of each block.

4. EXPERIMENTAL RESULTS

To compare the performance of the aforementioned four methods, we create an image database composed of 500 images for use in our experiment. The image resolution is of size 256×256. The content of those images includes landscape, buildings, flowers, human, animals, and so on. For each image, we randomly copy a region of size 81×81 and paste it to another location to form a tempered image. Since the detection results depend somewhat on the content of image and the region selected, we conduct the experiment for all the images and use the average value for comparisons. The following parameters are preset: $b = 256$, $Nn = 1$, $Nd = 10$, $Nc = 26$, and $Nt = 26$.

To compare the robustness of the four methods with respect to JPEG compression and Gaussian blurring, we use XNview (a software for viewing and converting graphic files, which is a freeware available at http://www.xnview.com), to accomplish the JPEG compression and Gaussian blurring. When conducting the JPEG compression, XNview allows us to choose the compression ratio between 0 and 100. The smaller the number, the smaller the output file will be. When conducting the Gaussian blurring, XNview allows us to choose the Gaussian blurring filter size. The larger the size, the more the given image will become blurred.

Fig. (**4**) shows an example of a copy-cover forgery, in which the tampering consists of copying and pasting a region to cover a significant content.

Fig. (4). (**a**) The original image, and (**b**) the copy-cover forgery image.

Fig. (**5**) shows the output images of copy-cover forgery detection when the given image is compressed with JPEG quality 50, where (a)-(d) are respectively obtained by PCA, DCT, spatial domain, and statistical domain detection methods.

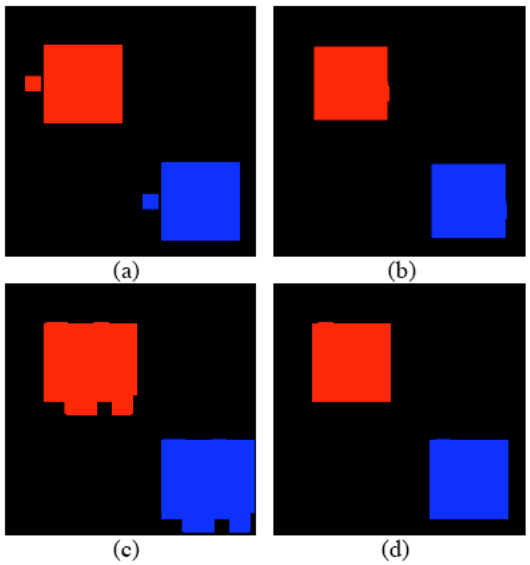


Fig. (5). Output of copy-cover forgery detection by (**a**) PCA, (**b**) DCT, (**c**) spatial domain, and (**d**) statistical domain detection methods. Note that the matched blocks are colored by two colors, red and blue.

Fig. (**6**) shows the output copy-cover forgery detection when the given image is corrupted by Gaussian blurring of block size 7.

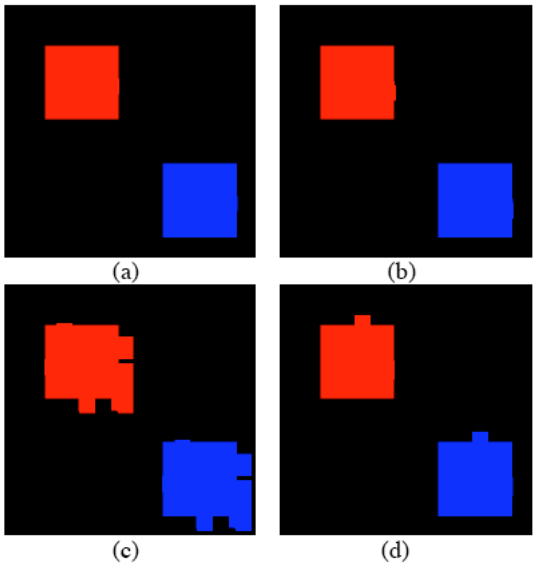
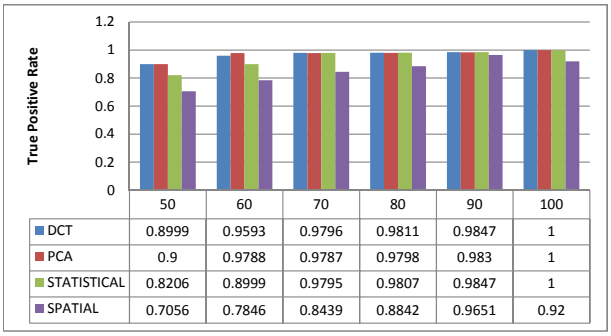


Fig. (6). Output of copy-cover forgery detection by (**a**) PCA, (**b**) DCT, (**c**) spatial domain, and (**d**) statistical domain detection methods when the given image is corrupted by Gaussian blurring with block size 7.

4.1. Robustness to JPEG Compression

Since most images available are JPEG compressed, we apply JPEG compression ratios from 50 to 100 to compress the test images for comparing the robustness of the four methods under JPEG compression. The obtained true positive rates related to JPEG quality are shown in Fig. (**7**). We

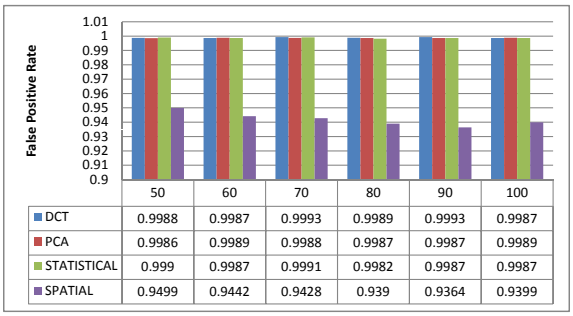
observe that the performances of DCT domain and PCA domain methods are very similar, and are better than those of spatial domain and statistical domain methods. Moreover, the statistical domain method is slightly better than the spatial domain method.



	50	60	70	80	90	100
DCT	0.8999	0.9593	0.9796	0.9811	0.9847	1
PCA	0.9	0.9788	0.9787	0.9798	0.983	1
STATISTICAL	0.8206	0.8999	0.9795	0.9807	0.9847	1
SPATIAL	0.7056	0.7846	0.8439	0.8842	0.9651	0.92

Fig. (7). True positive rates with respect to JPEG compression ratio.

The false positive rates with respect to JPEG quality are shown in Fig. (**8**). We observe that the performance of DCT domain, PCA domain, and statistical domain methods are very similar. However, the performance of spatial domain is worse than the other three.



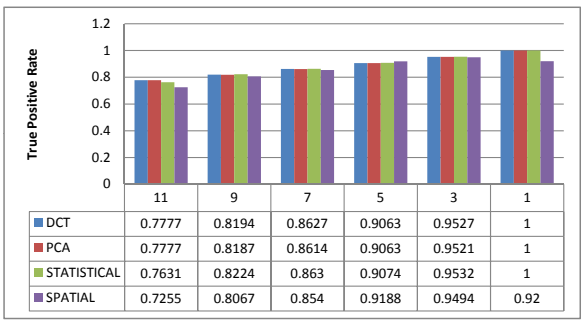
	50	60	70	80	90	100
DCT	0.9988	0.9987	0.9993	0.9989	0.9993	0.9987
PCA	0.9986	0.9989	0.9988	0.9987	0.9987	0.9989
STATISTICAL	0.999	0.9987	0.9991	0.9982	0.9987	0.9987
SPATIAL	0.9499	0.9442	0.9428	0.939	0.9364	0.9399

Fig. (8). False positive rates with respect to JPEG compression ratio.

4.2. Robustness to Gaussian Blurring

Since copy-cover image forgery will produce two identical regions in an image, one method to conceal the forgery is to apply Gaussian blurring on the composite image to conceal the forgery. We apply Gaussian blurring with different block sizes from 1×1 to 11×11 on the test images and then perform the detection. Note that the image using 1×1 Gaussian blurring is the same as the original image.

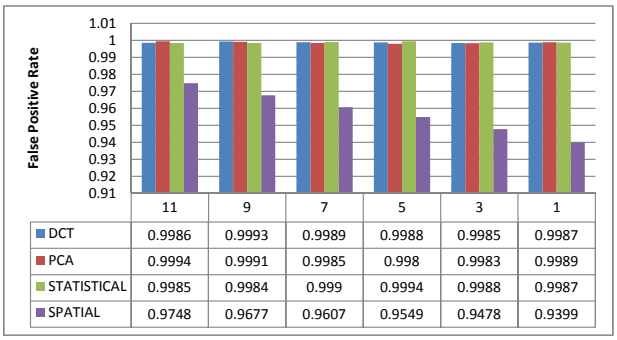
The true positive rates with respect to Gaussian blurring are shown in Fig. (**9**). We observe that the performances of DCT, PCA, and statistical domain methods are similar, which are slightly better than that of the spatial domain method.



	11	9	7	5	3	1
DCT	0.7777	0.8194	0.8627	0.9063	0.9527	1
PCA	0.7777	0.8187	0.8614	0.9063	0.9521	1
STATISTICAL	0.7631	0.8224	0.863	0.9074	0.9532	1
SPATIAL	0.7255	0.8067	0.854	0.9188	0.9494	0.92

Fig. (9). True positive rates with respect to Gaussian blurring.

The false positive rates with respect to PSNR are shown in Fig. (**10**). We observe that the false positive rate of the spatial domain method is the lowest. The false positive rates of DCT, PCA, and statistical domain methods are similar.



	11	9	7	5	3	1
DCT	0.9986	0.9993	0.9989	0.9988	0.9985	0.9987
PCA	0.9994	0.9991	0.9985	0.998	0.9983	0.9989
STATISTICAL	0.9985	0.9984	0.999	0.9994	0.9988	0.9987
SPATIAL	0.9748	0.9677	0.9607	0.9549	0.9478	0.9399

Fig. (10). False positive rates with respect to Gaussian blurring.

The experiment is performed on a DELL notebook computer with a 1.70GHz Intel Pentium Mobile Processor and 512 MB of RAM running Windows XP™. The program is coded in Matlab™. For a given image of size 256x256, the average running time of the four methods is shown in Table **1**.

Table 1. Running Time of the Four Detection Methods

Method	Running Time (second)
DCT domain	29.8594
PCA domain	20.5313
Spatial domain	17.0156
Statistical domain	20.8281

5. CONCLUSIONS

In this paper, we discuss the techniques of watermarking for authentication and the four methods for copy-cover identification, including PCA, DCT, spatial domain, and statistical domain. We evaluate their effectiveness and sensitivity under the influences of Gaussian blurring and lossy JPEG compressions. We conclude that the PCA domain method outperforms the other methods in terms of time complexity and detection accuracy. Our future work is to extend the capability of copy-and-paste modification from different images. Furthermore, we will explore more complicated spatially-distributed copy-and-paste modification; that is, instead of copying a consecutive area, one may copy one pixel here and another pixel there in a random-like distribution.

REFERENCES

[1] K. Hafner, "The camera never lies, but the software can", *New York Times*, 2004.

[2] F. Y. Shih, *Digital Watermarking and Steganography: Fundamentals and Techniques*. Boca Raton, FL: Taylor & Francis Group, CRC Press, 2007.

[3] Y. Wu, and F. Y. Shih, "Digital watermarking based on chaotic map and reference register", *Pattern Recognition*, vol. 40, pp. 3753-3763, 2007.

[4] P. Bas, J. M. Chassery, and B. Macq, "Image watermarking: an evolution to content based approaches", *Pattern Recognition*, vol. 35, pp. 545-561, 2002.

[5] C. Y. Lin, and S. F. Chang, "A robust image authentication method surviving JPEG lossy compression", *in SPIE Conf. Storage and Retrieval of Image/Video Database*, 1998, pp. 296-307.

[6] Y. Yuan, H. Decai, and L. Duanyang. "An integer wavelet based multiple logo-watermarking scheme", *in First International Multi-Symposiums on Computer and Computational Sciences*, 2006, pp. 175-179.

[7] D. Huang, Y. Yuan, and Y. Lu, "Novel multiple logo-watermarking algorithm based on integer wavelet", *Chinese Journal of Electronics*, vol. 15, pp. 857-860, 2006.

[8] H. T. Wu, and Y. M. Cheung, "Reversible watermarking by modulation and security enhancement", *IEEE Transaction on Instrumentation and Measurement*, vol. 59, pp. 221-228, 2010.

[9] N. K. Kalantari, S. M. Ahadi, and M. Vafadust, "A robust image watermarking in the ridgelet domain using universally optimum decoder", *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 20, pp. 396-406, 2010.

[10] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "Reversible image watermarking using interpolation technique", *IEEE Transaction on Information Forensics and Security*, vol. 5, pp. 187-193, 2010.

[11] X. Kang, H. Jiwu, and Z. Wenjun, "Efficient general print-scanning resilient data hiding based on uniform log-polar mapping", *IEEE Transaction on Information Forensics and Security*, vol. 5, pp. 1-12, 2010.

[12] B. Mahdian, and S. Saic, "Detection of copy-move forgery using a method based on blur moment invariants", *International Journal of Forensic Science*, vol. 171, pp. 180-189, 2007.

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

[14] A. C. Popescu, and H. Farid, "Exposing digital forgeries by detecting duplicated image regions", Technical Report, TR2004-515, Dartmouth College, 2004.

[15] S. Ju, J. Zhou, and K. He, "An authentication method for copy areas of images", *in Intl. Conf. Image and Graphics*, 2007, pp. 303-306.

[16] I. T. Jolliffe, *Principal Component Analysis*, Springer, 2002.

[17] P. Yip, and K. Rao, *Discrete Cosine Transform: Algorithms, Advantages, and Applications*, Academic Press, 1990.

Received: January 09, 2010 Revised: March 31, 2010 Accepted: April 05, 2010