Available online at www.sciencedirect.com
ScienceDirect
Computer Standards & Interfaces 30 (2008) 132–136
COMPUTER STANDARDS & INTERFACES
www.elsevier.com/locate/csi

Detecting fake images using watermarks and support vector machines

Wei Lu [a,b,*], Fu-Lai Chung [c], Hongtao Lu [b], Kup-Sze Choi [c]

[a] Department of Computer Science, Sun Yat-sen University, Guangzhou 510275, China
[b] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200030, China
[c] Department of Computing, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

Received 11 May 2006; received in revised form 24 April 2007; accepted 10 August 2007
Available online 19 August 2007

Abstract

With the great convenience of computer graphics and digital imaging, it becomes much easier to alter the content of images than before without any visually traces to catch these manipulations, i.e., many fake images are produced whose content is feigned. Thus, the images can not be judged whether they are real or not visually. In order to detect fake images, this paper proposes a detection scheme, which, firstly, uses watermarks to locate the alteration, and then, uses support vector machine (SVM) as a classifier to make a binary decision on whether an image is fake or real. The experimental results also demonstrated the effectiveness of the proposed scheme.
© 2007 Elsevier B.V. All rights reserved.

Keywords: Fake image detection; Watermarks; Support vector machines

1. Introduction

In recent years, it is much easier to alter the content of digital images than before using the popular image editing computer software. Some of these images are irrational with respect to their content, making their fakery obvious to human. However, many of the altered images can not be easily determined as real or fake. Fake images are loosely defined as images that do not express the original content [1]. In other words, what human saw is not necessarily believable. Like it or not, fake images are everywhere, such as those in movies, advertisements, etc. There is no general model to detect them, since the production of fake images leaves no visual clues of fakery. Fig. 1 gives some examples of fake images, which show that fake images are consistent in their signal characteristic with real images, and only their contents are exaggerated. To the fake images that are produced for the sake of cheating, especially for political advantage, they can barely be distinguished only based on their content.

Some similar concepts with image fakery have been introduced in the past few years, such as digital forgery [2] and image splicing [3]. In [2] digital forgeries are referred to the manipulation and alteration of digital images, and method to detect traces of resampling is proposed to expose digital forgeries. Furthermore, some other statistical tools for detecting digital forgeries are also proposed and analyzed in [4], including techniques employing the inconsistences in digital camera imaging techniques [5], digital image sampling techniques, the direction of point light [6], principal component analysis [7] and higher-order wavelet statistics [8]. In [3] an image splicing model is proposed to combine different objects in images into a new image. Aiming at detecting spliced images a statistical model using bicoherence feature has been proposed in [9], which was originally designed for detecting human speech signal.

Practically, it is hard to say whether an image is real or fake only from viewing the content of the image, because the purpose of creating fake images is to alter the content of an image by adding, removing or replacing some objects in the image, so that the altered image looks like a real image. For Fig. 1(a), it is illogical for a flying hawk hanging a bomb, so it may be easily concluded that this image is fake. The same conclusion can be applied to Fig. 1(b). Unfortunately, most of the fake images can not be distinguished visually and appropriate methods should be developed to detect them. In this paper, a detection scheme using watermarks and support vector machine (SVM)

* Corresponding author.
E-mail addresses: luwei3@mail.sysu.edu.cn (W. Lu), cskchung@comp.polyu.edu.hk (F.-L. Chung), lu-ht@cs.sjtu.edu.cn (H. Lu).

0920-5489/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.csi.2007.08.002

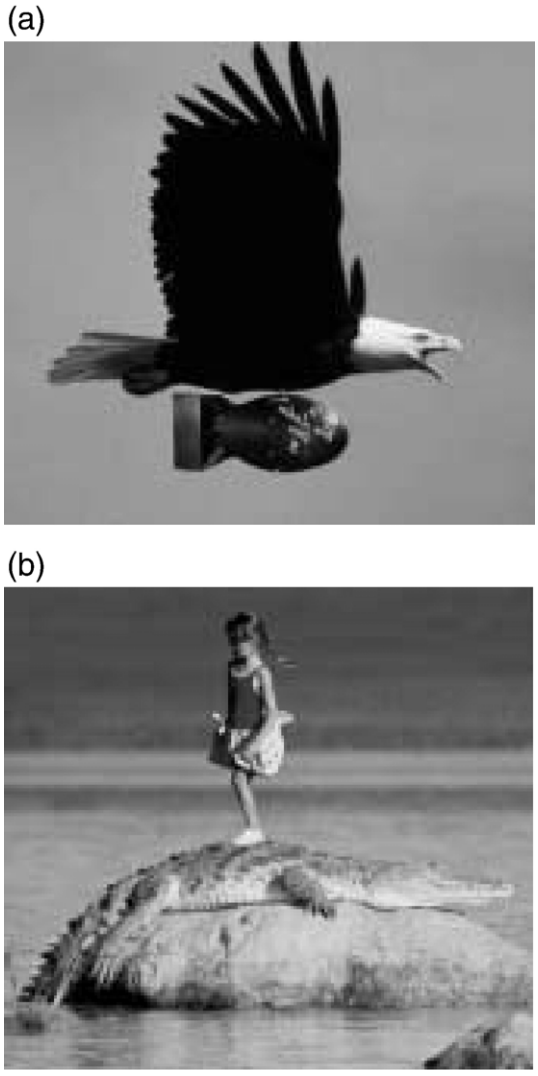


Fig. 1. Examples of fake images.

is proposed. Firstly, a watermark is inserted into the original image, and the altered area can then be detected by extracting the watermark and comparing it with the original watermark, a classifier using SVM is then used to detect whether the input images are fake or not based on the constructed features.

2. Feature construction

In order to carry out fake image classification using SVM, first of all, we need to construct a set of the features x_i and it is described in detail in this section. Firstly, a fragile digital watermarking scheme is introduced as an assistant, which is used to find out the alteration of images. Then, the feature vector is constructed by convoluting two vectors which are obtained from the detected alteration matrix.

2.1. Watermarking of images

In our fake image detection scheme, the altered area in an image should be detected first. Here, we take use of a least significant bit (LSB) based fragile watermarking scheme [10] to watermark the original image, and then the watermarked image is open to public. Digital watermarking has been developed over 10 years. The main contributions are the robust watermarking which emphasizes the existence of the watermark. On the other hand, the fragile watermarking received less attention and it concerns with how to achieve the sensitivity to the image's alteration. With the assistant of fragile watermarking, even a slight alteration to the test images can be detected.

In [10], a fragile watermarking scheme is introduced, where a watermark W with the same size of the original image, i.e., $m \times n$, is constructed to embed into the LSB plane of the original image I. Each watermark element is inserted into the LSB of each image pixel. Then, a watermarked image I_w is obtained. When the watermark is extracted form the test image I', there will be difference between the extracted watermark W_e and the original watermark W, if the watermarked image is altered. Based on the difference, a matrix can be obtained as follow:

$$A = \mathrm{XOR}(W, W_e) \tag{1}$$

where A is a matrix composed of 0 and 1 with size $m \times n$. It can be easily concluded that when $A(s,t)=1$, the pixel $I'(s,t)$ in the test image is altered. On the other hand, when $A(s,t)=0$, it is not altered. Here (s,t) is the pixel index with $s=1,2,\ldots, m$ and $t=1,2,\ldots, n$. In Fig. 2, a sample detection process in shown. Here, the white pixels in Fig. 2(b) denotes the altered pixels.

2.2. Construction of feature vectors

Once we get the matrix A, two column vectors $U=\{u_1,u_2,\ldots, u_m\}^T$ and $V=\{v_1,v_2,\ldots, v_n\}^T$ can be obtained as follows:

$$u_s = \frac{1}{n}\sum_{t=1}^{n} A(s,t), \quad s = 1,2,\ldots,m. \tag{2}$$

$$v_t = \frac{1}{m}\sum_{s=1}^{m} A(s,t), \quad t = 1,2,\ldots,n. \tag{3}$$

which show the number of the altered pixels along the row and the column of A respectively. It is obvious that U and V reveal the altered area in the test image from two directions.

In the proposed scheme, we use two methods to construct the feature vector. The first one is to joint the two vectors U and V as follows:

$$\begin{aligned} x_i &= U + V = \{u_1, u_2, \ldots, u_m, v_1, v_2, \ldots, v_n\}^T \\ &= \{x_i(1), x_i(2), \ldots, x_i(m+n)\}^T \end{aligned} \tag{4}$$

where + denotes the joint operator. The length l of the feature vector is $m+n$. The other way to construct the feature vector is to convolute U and V, so that a feature vector x_i with length $l=m+n-1$ is obtained as follows:

$$x_i = U \otimes V \tag{5}$$

Fig. 2. (a) A fake version of Lenna image. (b) Area detected with alteration (white pixels).

where $\otimes$ denotes the convolution operator and the k-th element of the feature vector x_i can be obtained from the convolution equation as follows:

$$x_i(k) = \sum_{j=1}^{m} u_j v_{k+1-j} \tag{6}$$

Fig. 3 shows a sample of U, V and $U \otimes V$ curves. Once the feature vector is constructed, the training set can be obtained as $T = \{x_i, y_i\}$, where y_i is the fakery indicator of the i-th image, i.e., either -1 or 1.

3. Support vector classification

Support vector machine is a technique for universal data classification. In recent years, SVMs have been used for many applications, such as pattern recognition, industrial engineering, digital watermarking and so on. Generally, SVMs are deemed to be easier and better to use than traditional neural network models.

The ideal of SVM is to construct a mapping model from input data to output data which are also defined as features for input data and targets for output data. Generally speaking, there are two data sets in classification, i.e., training data and testing

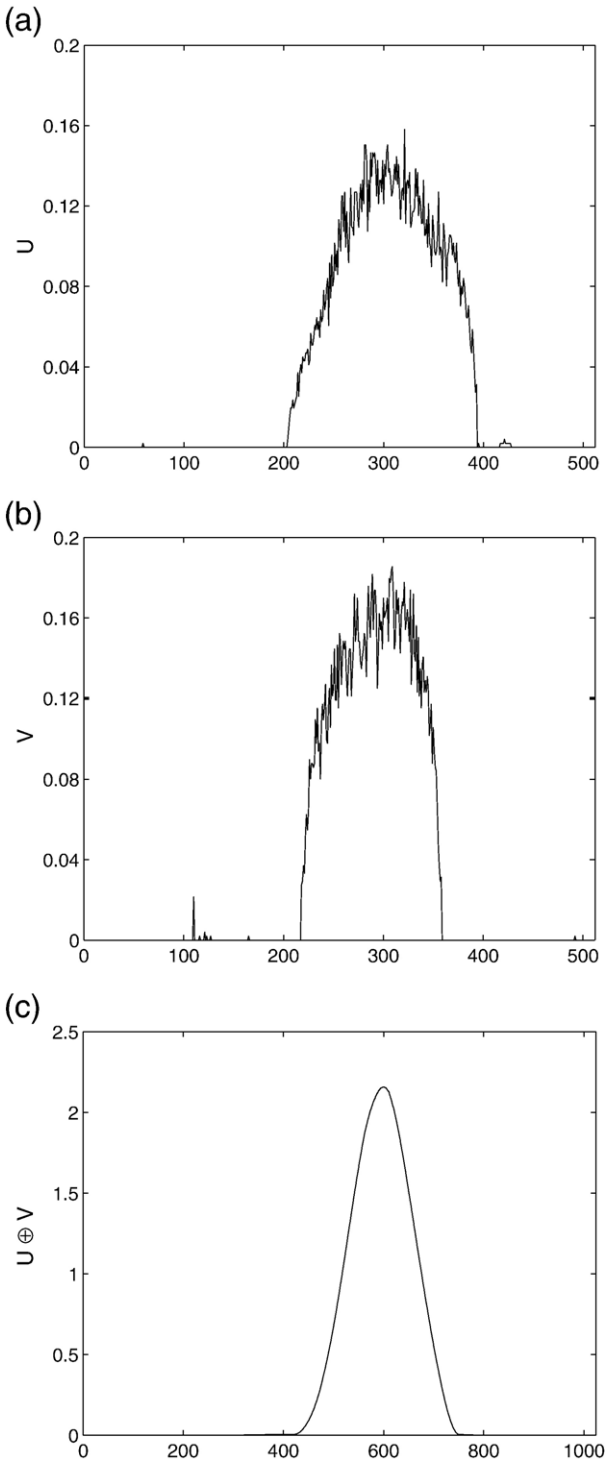


Fig. 3. The curves of U, V and their convolution $U \otimes V$ are shown in (a), (b) and (c) respectively.

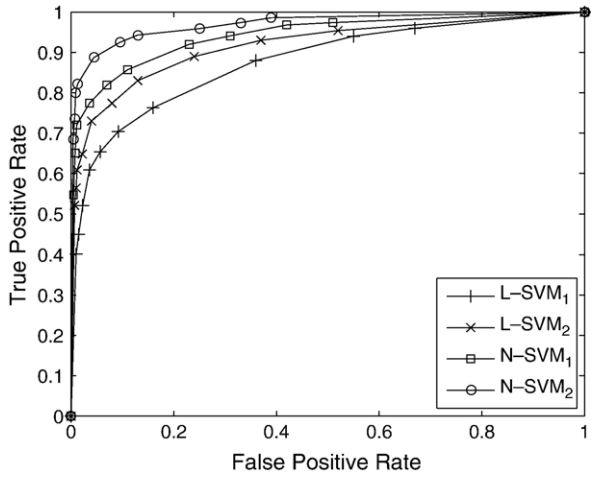


Fig. 4. The ROC curves between the false positive rate and the true positive rate for the proposed SVM classifiers.

data. Each training data contains several features and one target. After SVM learns using the training data, SVM can produce a model to predict the corresponding target of the test data. Given a training data set $\{(x_I,y_i)|i=1,2,\ldots, l\}$, where $x_i \in R^d$ and $y_i \in \{1,-1\}$, SVM is to resolve the following optimization problem:

$$\min \quad \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i$$

$$\text{subject to} \quad (w^T \Phi(x_i) + b) y_i \geq 1 - \xi_i,$$

$$\xi_i \geq 0. \tag{7}$$

where x_i are the feature vector in the training data set which is mapped into a higher dimensional space $\mathcal{H}$ by the function $\Phi(\cdot)$, and $C>0$ is the penalty parameter of the error function. Then, SVM finds a linear separating hyperplane with the maximal margin in the space $\mathcal{H}$. If the hyperplane exists, the parameters in Eq. (7) can be calculated, and the hyperplane $w^T\Phi(x_i)+b$ can be obtained. Moreover, $K(x_i,x_j)=\Phi(x_i)^T\Phi(x_j)$ is defined as the kernel function. By using different kernels, SVMs can be specified for different uses. Practically, there are four basic kernels which are often used, i.e., linear kernel, polynomial kernel, radial basis function (RBF) kernel and sigmoid kernel. Especially, when $\Phi(x_i)=x_i$, the kernel becomes a linear kernel $K(x_1,x_2)=x_1^T x_2$, and $R^d \equiv \mathcal{H}$, and thus, if a hyperplane can be obtained, it becomes a linear SVM. Furthermore, if $\xi_i=0 \forall_i$, it becomes a linear separable SVM.

In our fake image detection problem, the training set $T(x_i,y_i)$ consists of the feature vectors x_i and the target y_i, where $y_i=1$ for real images, and $y_i=-1$ for fake images. After the SVM is trained using these training data, the parameters in Eq. (7) can be obtained. Then, a test example, z, can be classified by determining on which side of the separating hyperplane it lies. Assuming that it is a separable case, specifically, if $w^T\Phi(z)+b \geq 0$, the example is classified as a real image, otherwise the example is classified as a fake image.

4. Results and discussions

In this section, we report the experiments carried out to validate the effectiveness of the proposed fake image detection scheme. In order to train the SVMs, an image database consisting of 1000 real images and 2000 fake images of size 256×256, 512×512, 1024×1024 and others is used. For the image database construction, firstly, we collected 1000 real JPEG images, which includes different well-proportioned sizes. Then, these real images are watermarked using the proposed scheme in [10]. Based on the watermarked images, we build 2000 fake versions of these images by means of popular image processing softwares, Photoshop and GIMP. As is shown in Fig. 2(a), our database consists of also a fake Lenna image. We have simulated the SVMs using different kernels, including linear SVM (L-SVM) and nonlinear SVM with Gaussian RBF kernel (N-SVM) such that $K(x_i,x_j)=\exp(-\|x_i-x_j\|^2/2\sigma^2)$. Furthermore, the SVMs take use of the two feature vectors proposed in Section 2, and they are labeled as L-SVM$_1$ and N-LVM$_1$ for the feature vector in Eq. (4), and L-SVM$_2$ and N-LVM$_2$ for the feature vector in Eq. (6). In the training stage, 500 real images and 1000 fake images are randomly selected to fed to the proposed SVMs, the other images are used to test the trained SVMs.

Fig. 4 shows the ROC curves between the false positive rate and true positive rate for different SVMs, where the false positive rate is the percentage of real images that are incorrectly classified as fake, and the true positive rate is the percentage of fake images that are correctly classified as fake. The larger for the area under the curve, the better for the performance of the corresponding detection schemes. Furthermore, Table 1 gives the experimental data of the proposed SVM classification schemes for images with different size. It can be concluded that the performance for nonlinear SVMs using RBF kernel is better than that for linear SVMs, and the performance for the SVMs using the convolution type feature vector in Eq. (6) is better than the methods in Eq. (4) under the proposed classification scheme. From Fig. 3, we can see that the positions of each curve's peak are movable with different fake area in different images, so, it is highly nonlinear between the input vector x_i and the output decision value y_i for the applied SVMs. Thus, dut to complex nonlinear classification capability, nonlinear SVMs are better than linear SVMs. What's more, the features constructed using Eq. (6) are better than that using Eq. (4), we think the convolution between the altered pixel vectors in the direction of rows in Eq. (3) and columns in Eq. (2) can increase the robustness and stability of digital fakery, while the joint operation can not combine the features from the direction of rows and columns.

Table 1
The classification accuracy (%) (true positive rate) with 1.0% false positive rate with different image size

Image Set	L-SVM$_1$	N-SVM$_1$	L-SVM$_2$	N-SVM$_2$
256×256	40.3	69.3	57.2	85.7
512×512	42.8	71.6	53.5	83.4
1024×1024	39.3	72.3	56.9	83.8
Others	44.1	67.1	54.3	84.1

In the proposed fake detection scheme, the detection result depends on the constructed feature vector x_i. In order to further test the performance of the proposed scheme, we analyzed the relation between the images and the detection results, and found that the true negative detection mostly concentrates on the images with small fake area, and it is irrelevant to the image size, so it can be concluded that the proposed detector becomes more sensitive with smaller fake area, and it is partially due to the robustness of watermarking and the fake intensity.

We also assigned the training images with random outputs of 1 and − 1, where half of the images are randomly assigned to the fake images and the others are the real images. Then we trained the N-SVM$_2$ using the feature vector in Eq. (6) and then tested it. We found that the classification accuracy is 42.6% for the real images with the false negative rate 1.2%, which is much worse than the case when the correct outputs are assigned. This indicates that the proposed features and SVM classifiers are based on general characteristics for real and fake images.

5. Conclusions

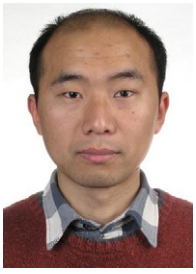
In this paper, we have proposed a watermark and SVM based classification to detect fake images. A LSB-based watermarking is first applied to detect the altered area, and based upon which, a feature vector is constructed for the SVM classification. Thus, fake images can be detected based on the output of the SVMs. Future works include the performance enhancement of the proposed detection scheme using more sophisticated features and the watermark-free fake image detection.

Acknowledgments

This work is supported by the Scientific Research Foundation for Young Teachers in Sun Yat-sen University, the Hong Kong PolyU ICRG grant under project A-PG49, NSFC under project no. 60573033, and Program for New Century Excellent Talents in University (no. NCET-05-0397).

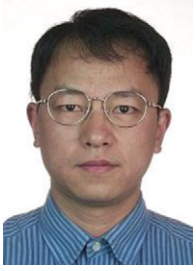
References

[1] R.D. Fiete, Photo fakery. URL http://oemagazine.com/fromTheMagazine/jan05/photofakery%.html.
[2] A.C. Popescu, H. Farid, Exposing digital forgeries by detecting traces of resampling, IEEE Transactions on Signal Processing 53 (2) (2005) 758–767.
[3] T.-T. Ng, S.-F. Chang, A model for image splicing, IEEE Int. Conf. on Image Processing (ICIP), vol. 2, 2004, pp. 1169–1172.
[4] A. Popescu, H. Farid, Statistical tools for digital forensics, 6th International Workshop on Information Hiding, Vol. 3200, Toronto, Cananda, 2004, pp. 128–147.
[5] A. Popescu, H. Farid, Exposing digital forgeries in color filter array interpolated images, IEEE Transactions on Signal Processing 53 (10) (2005) 3948–3959.
[6] M. Johnson, H. Farid, Exposing digital forgeries by detecting inconsistencies in lighting, ACM Multimedia and Security Workshop, New York, 2006, pp. 1–9.
[7] A.P. Farid, Exposing Digital Forgeries by Detecting Duplicated Image Regions, Tech. Rep. TR2004-515, Department of Computer Science, Dartmouth College, 2004.
[8] H. Farid, 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, pp. 16–22.
[9] T.-T. Ng, S.-F. Chang, Q. Sun, Blind detection of photomontage using higher order statistics, IEEE Int. Symposium on Circuits and Systems (ISCAS), vol. 5, 2004, pp. 688–691.
[10] H. Lu, R. Shen, F.-L. Chung, Fragile watermarking scheme for image authentication, Electronics Letters 39 (12) (2003) 898–900.



Wei Lu was born in China in 1979. He received his B.Sc degree in automation from Northeast University, China in 2002 and his M.Sc and Ph.D. degrees in computer science from Shanghai Jiao Tong University, China in 2005 and 2007 respectively. He joined the Department of Computer Science, Sun Yat-sen University, China in 2007. His current research interests include signal and image processing, pattern recognition, information forensics and security, multimedia watermarking, steganography.

Fu-lai Chung received his B.Sc. degree from the University of Manitoba, Canada in 1987 and his M.Phil. and Ph.D. degrees from the Chinese University of Hong Kong in 1991 and 1995 respectively. He joined the Department of Computing, the Hong Kong Polytechnic University, in 1994, where he is currently an associate professor. He has published widely in the areas of soft computing, data mining, machine intelligence, and multimedia and his current research interests include financial data mining, fake image processing and novel computational intelligence techniques.



Hongtao Lu received his Ph.D. degree in electrical engineering from Southeast University, Nanjing, China, in 1997. He is currently a professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. His research interests include neural networks, chaos and complex networks, image processing and pattern recognition.

Kup-Sze Choi received his Ph.D. degree in Computer Science and Engineering from the Chinese University of Hong Kong in 2004. He is an assistant professor with the Department of Computing, Hong Kong Polytechnic University. His current research interests include computer graphics, virtual reality, physics-based modeling, computer-assisted surgery and virtual-reality applications in medicine.