EFFECTIVE IMAGE SPLICING DETECTION BASED ON IMAGE CHROMA

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

A color image splicing detection method based on gray level co-occurrence matrix (GLCM) of thresholded edge image of image chroma is proposed in this paper. Edge images are generated by subtracting horizontal, vertical, main and minor diagonal pixel values from current pixel values respectively and then thresholded with a predefined threshold T. The GLCMs of edge images along the four directions serve as features for image splicing detection. Boosting feature selection is applied to select optimal features and Support Vector Machine (SVM) is utilized as classifier in our approach. The effectiveness of the proposed method has been demonstrated by our experimental results.

***Index Terms*—** image chroma, co-occurrence matrix, boosting feature selection, image splicing detection, SVM

1. INTRODUCTION

In the current digital age, creating and manipulating digital images are easy and simple by using digital processing tools which are widely available from the Internet. However, some people take this opportunity to do something wrong. Many tampered images emerge in news coverage, scientific experiments and even legal evidences. Therefore, we can no longer take the authenticity of images for granted. The need for image tampering detection makes image forensics a very important research issue. Image splicing is a fundamental operation used in digital image tampering. It is a copy-and-paste operation of image regions from one image onto the same or another image without performing post-processing such as smoothing [1]. Even without post-processing, the artifacts introduced by image splicing may be almost imperceptible [2]. Therefore, the detection of image splicing is a preliminary but desirable study for image forensics.

Generally speaking, there are two approaches of image forgery detection: active [3, 4] and passive [5, 1] detection. Active approach requires pre-processing (e.g. watermark embedding) when generate image or before distribute image. However, many of the image capture devices are lack of this functionality, which makes the active approach not universal. The passive approach does not need this operation, however, and could make analysis on various images based on supervised learning. Hence, it gains more attention and becomes a hot research topic. In this paper we focus on passive image splicing detection based on machine learning.

Recently, several methods for detecting image splicing were proposed. Ng et al. [1] employed a higher order moment spectrum, bicoherence, as features to detected the presence of the abrupt discontinuities in a suspected image which can be considered as a clue for detecting spliced images. However, the detection rate was not high. In [6], Johnson and Farid developed a technique of image tampering detection by detecting the inconsistency of lighting in an image. But the method may fail when source images which are used for tampering are taken under approximately similar lighting conditions. Hsu and Chang employed camera characteristic consistency to detect spliced images in [7]. However, this method needs to label the suspicious regions of image manually before making decision. In [2] Chen et al. proposed a splicing detection method based on statistical features of characteristic functions and phase congruency which is sensitive to sharp transitions such as lines, edges and corners caused by splicing operation. But the detection rate was also not high and they are time consuming for feature extraction. Some effective features based on Markov transfer matrices were introduced for image splicing detection in [8], and the experimental results were satisfied.

Most of the methods mentioned above are used in grayscale image or luminance component of color image. However, these will neglecte useful information for color image splicing detection. Surely, we can use R, G and B channels respectively for splicing detection, but it also neglect the relationship among these channels. So far as we know, there are few papers specially for color image splicing detection (Chen et al used HSV color space to identify computer graphics [9]), and the only public available image dataset for image splicing detection [10] consist of grayscale images. Color image splicing detection is more meaningful than gray image splicing detection since most images are color ones in our real word. Therefore, in this paper, we proposed an effective color image splicing detection method by using image chroma which can better reflect the splicing introduced trace.

The rest of this paper is organized as follows. Our proposed features for splicing detection are introduced in Section 2. In Section 3, the experimental results are reported and some analyses are given. Conclusion is drawn in Section 4.

2. FEATURE EXTRACTION

As we know, image splicing often changes image structure, which means sharp edges are usually introduced by splicing. How to distinguish these introduced edges from ”original” image edges would be an important clue in image splicing detection. In this section, we will introduce how to extract and select features, which are effective to splicing detection, based on the observation of image edges in chroma channel.

2.1. Color space

First of all, we have to introduce an image format used in this paper. YCbCr is a family of color spaces just like RGB. Y is the luminance component and Cb and Cr are the blue-difference and red-difference chroma components. Cb (or Cr) component has little image content while most of image content is preserved in Y component. Here, we will give an example shown in Figure 1. When come to color image, most of splicing detection methods only use luminance component of it, and chroma component is dropped. However, we find the image chroma is very useful for color image splicing detection. Since human are more sensitive to luminance than to chroma, even if spliced image looks natural to human, some unnatural clues will be left in chroma channel. Therefore, we could make use of the chroma information of the image for splicing detection.

Fig. 1. An example of spliced images (The region of bird is spliced onto the authentic image). (a) Color image (RGB format), (b) Y component, (c) Cr component , (d) Cb component

Here we take Figure 1 as an example to further explanation. In Figure 1, the region of bird is spliced onto the authentic image. When we carefully look into the bird's contour in the Y, Cr and Cb component, we find that the plenty image details cover up the splicing introduced edges in Y component, while in Cr (or Cb) component the bird's contour (which presents the spliced region) is sharper than other objects' in the image. The splicing caused edges would be more detectable in chroma component. Hence, we will introduce edge image in following.

2.2. Edge image

In order to weaken the effect of residual image content in Cb (or Cr) component, edge image is introduced in this part. Zou et al. [11] used predict-error to weaken the effect of image content for steganalysis. Similar to predict-error image, edge image is an image that only has edge information of an original image. An edge image can be generated by applying an edge detector (e.g Sobel, LoG or Canny) onto an original image. In this paper, we adopt a simple detector. Four edge images denoted by E_h, E_v, E_d and E_{-d} are generated by image predict-errors listed as follows:

$$\begin{aligned} E_h(i,j) &= |x(i+1,j) - x(i,j)| \\ E_v(i,j) &= |x(i,j+1) - x(i,j)| \\ E_d(i,j) &= |x(i+1,j+1) - x(i,j)| \\ E_{-d}(i,j) &= |x(i+1,j-1) - x(i,j)| \end{aligned} \tag{1}$$

where $x(i,j)$ indicates the gray value of the pixel at location (i,j). Denoise operation should be adopted before calculating predict-errors, because predict-errors are sensitive to noise. An example of E_d is shown in Figure 2 (The area of zebra is spliced into the authentic image).

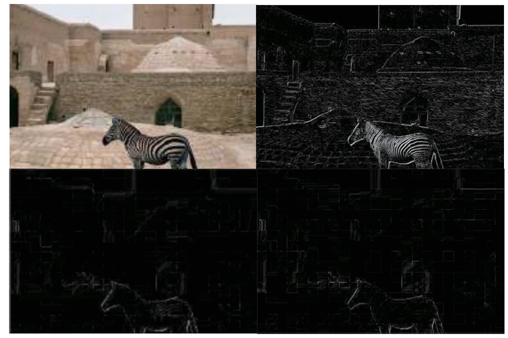

Fig. 2. An example of spliced images and E_d of **Y**, **Cb** and **Cr** components from top to down and left to right, respectively. The area of zebra is spliced into the authentic image.

From Figure 2, we find that in the edge image of Y component, most of the edges including original image edges and splicing introduced edges are clearly seen. However, in the edge images of Cb (or Cr) component, the original image edges are much more smooth than the splicing introduced edges (say the contour of zebra) and even some of them are disappeared. Hence, the edge images of Cb (or Cr) component are much more sensitive to the splicing introduced edges than the edge images of Y component. In another word, if a test image is an authentic one, its edge images of Cb (or Cr) component should be smooth without any obvious sharp edges. If the test image is a spliced one, its edge images of Cb (or Cr) component should contain some sharp edges (caused by splicing). Based on this observation, we make use of Cb (or Cr) component rather than Y component for further analysis.

2.3. Gray level co-occurrence matrix of thresholded edge image

The gray level co-occurrence matrix (GLCM) is a well-known representation for extracting second-order texture information from images. Here, we make use of GLCM in the edge image of Cb (or Cr) component for splicing detection. The co-occurrence matrix can be thought as an estimation of the joint PDF of gray-level pairs in an image. Since almost all of gray values in the edge image of Cb (or Cr) component are not big (about 95% of them are below 10), we need to threshold it with an reasonable value to reduce the size of GLCM that serves as feature. For this reason, a predefined threshold is adopted and edge images are adjusted according to the following rule:

$$e(i,j) = \begin{cases} e(i,j) & \text{e(i,j)} < \text{T} \\ T & \text{e(i,j)} \geq \text{T} \end{cases} \tag{2}$$

where $e(i,j)$ is the value of an edge image at location (i,j). After thresholding edge images, we can get four gray level co-occurrence matrices (the size of each matrix is $(T+1)*(T+1)$) of edge images in the following way:

- The element of GLCM of thresholed E_h denoted by CM_h is $P\left(E_h(i,j), E_h(i,j+1)\right)$.
- The element of GLCM of thresholed E_v denoted by CM_v is $P\left(E_v(i,j), E_v(i+1,j)\right)$.
- The element of GLCM of thresholed E_d denoted by CM_d is $P\left(E_d(i,j), E_d(i+1,j+1)\right)$.
- The element of GLCM of thresholed E_{-d} denoted by CM_{-d} is $P\left(E_{-d}(i,j), E_{-d}(i+1,j-1)\right)$.

These four matrices (CM_h, CM_v, CM_d, and CM_{-d}) are used to generate features by the way that each matrix is first transformed to a vector, and then cascaded them to form one feature vector. Therefore, the length of feature vector is $4*(T+1)*(T+1)$.

2.4. Boosting Feature Selection

A dimension reduction method is applied to our proposed feature in order to decrease the computational complexity of training and testing. Boosting Feature Selection (BFS) [12] proposed by Tieu et al. can be used to select optimal features to reduce the computational complexity. BFS can not only reduce dimension but also improve the detection accuracy. The effectiveness of BFS is verified by [13]. After D iterations, we can get D-dimensional new feature vectors, then they will be used for final classification.

3. EXPERIMENTAL RESULTS

3.1. Image dataset

The only available public image dataset for splicing detection is provided by DVMM, Columbia University [10]. This image dataset has 933 authentic and 912 spliced image blocks of size 128 x 128 pixels. However, all blocks come from grayscale images. Therefore, we have to construct color image dataset to test our proposed approach. We collected 500 authentic and 448 spliced images of size $384*256$ (or $256*384$). Some of authentic images are from *CorelDraw* image dataset and the Internet, others are taken by our digital cameras. All of them are JPEG format. They consist of scene, nature, architecture, article, people, animal, plant and texture images. We constructed the spliced images by the following ways:

- copy image region(s) from one image and paste onto the same image.
- copy image region(s) from one image (or some images) and paste onto another image.
- The spliced region(s) can be regular or irregular, interested or arbitrary.
- The spliced region(s) can undergo pre-processing (e.g. resize and rotation) or not.
- The sizes of spliced regions are various (big, medium or small) compared with the image size.

Some examples of our dataset are shown in Figure 3.

Fig. 3. Some examples of our image dataset. All images at top are authentic images and at bottom are spliced ones.

3.2. Classifier

Support Vector Machine (SVM) is an optimal and efficient classifier which is commonly used for machine learning systems. Since our work in this paper only focuses on feature extraction rather than the design of classifier, we utilize the LIBSVM [14] as the classifier in our experiment and a RBF kernel is chosen. Grid searching is used to select parameters for the classifier. All the experiments and comparisons are tested on the same database and the same classifier in this paper.

3.3. Detection Performance

In our experiment, we typically choose $D = 30, 50, 75, 100$ for BFS algorithm. The training samples were randomly selected from the image dataset (500 authentic and 448 spliced images) to train the classifier. We selected the training sample size to be 5/6 of images (417 authentic and 373 spliced) in order to get better model in training. The remaining images were used in testing. We performed 5 runs (by randomly selecting training samples each run) of RBF kernel SVM classifier with the parameters C and γ which are select by grid searching. In each run, plenty of experiments with different thresholds and different dimensions (Ds) were carried out. To prove the effectiveness of BFS, the experiments without BFS were also implemented. The best results (the highest test accuracy rates) were choose from these 5 experimental results and shown in Table 1.

Table 1. Experiment results of proposed method

Accuracy		$D=30$	$D=50$	$D=75$	$D=100$	no BFS
$T=7$	Y	69.0%	67.7%	69.0%	69.0%	64.6%
	C_b	84.8%	83.5%	82.9%	86.0%	87.3%
	C_r	87.3%	87.3%	89.2%	86.0%	88.0%
$T=8$	Y	66.5%	69.6%	67.1%	69.6%	66.5%
	C_b	84.2%	83.5%	86.1%	86.1%	88.6%
	C_r	87.3%	90.5%	88.6%	89.9%	89.9%
$T=9$	Y	62.7%	69.6%	66.5%	69.6%	57.6%
	C_b	82.9%	84.2%	86.7%	88.6%	86.7%
	C_r	88.0%	88.6%	87.3%	88.6%	84.2%

The dimension of "no BFS" is 256, 324 and 400 for $T = 7$, $T = 8$ and $T = 9$ respectively. From Table 1 we can find that features extract from Cb (or Cr) component perform much better than that from Y component. Though feature dimension is reduced, the test accuracy rate using reduced feature vectors are not worse. Therefore, the effectiveness of BFS has been proved. The highest accuracy rate 90.5% achieved on Cr component with threshold $T = 8$ and dimension $D = 50$.

Figure 4 shows the corresponding ROC curve of each component with $T = 8$ and $D = 50$. From Figure 4, we can also verify that the performance on Cb and Cr component are much better than on Y component.

To further testify the effectiveness of Cb (or Cr) component for splicing detection, we tested the method proposed by Chen et al. [2] on Y, Cb and Cr components of color images respectively under the same circumstance. The corresponding ROC curve of the detection from each component is shown in Figure 5. We can notice that the detection on Cb (or Cr) component is also more effective than Y component. Besides, from curves in Figure 4 and Figure 5, we

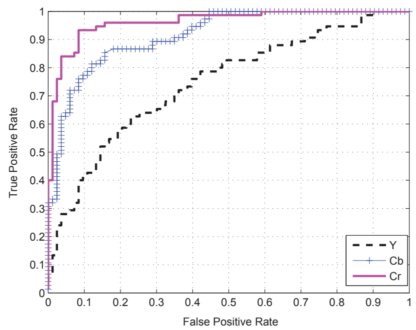

Fig. 4. ROC curve of each component using proposed method

also can find that our proposed method are better than the method proposed in [2].

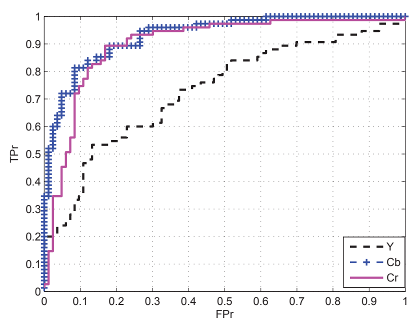

Fig. 5. ROC curve of each component using method proposed in [2]

Actually, in this paper, we only want to emphasize that the chroma channel (Cb or Cr channel) performs better than Y channel. However, if we combine these three channels' detection results by voting, the detection result should be higher. It will not be discussed here for lack of space.

4. CONCLUSION

This paper has proposed a passive color image splicing detection method based on the analysis of image chroma component. The experimental results have proved that the proposed features of Cb (or Cr) component are more effective than that of Y component. After feature extracting, feature selection (boosting feature selection) has been carried out in order to reduce feature dimensions. The detection accuracies using feature reducing were no worse than not using.

As we known, image splicing detection is the problem of weak signal detection in the background of strong signal. All that we did is to remove the strong signal (image content) at the same time to preserve the weak signal (splicing introduced edges). Therefore, after we got the edge image of Cb (or Cr) component, the problem has been changed to detection of signal in the background of weak signal which is an easier one to deal with.

Our proposed approach is a first attempt but primary study for color image splicing detection, and the experimental results have demonstrated the effectiveness of our method. We would further applying our method to other image forgery detection in our future work.

Acknowledgments.

This work is funded by research grants from the National Fundamental Research Program of China (Grant No.60603011) and the National Laboratory of Pattern Recognition. The authors also thank the anonymous reviewers for their valuable comments.

5. REFERENCES

[1] T.T. Ng, S.F. Chang, and Q. Sun, "Blind detection of photomontage using higher order statistics," in *IEEE International Symposium on Circuits and Systems*, 2004.

[2] W. Chen, Y.Q. Shi, and W. Su, "Image splicing detection using 2-d phase congruency and statistical moments of characteristic function," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, Feb 2007, vol. 6505.

[3] C. Rey and J.L. Dugelay, "A survey of watermarking algorithms for image authentication," *EURASIP Journal on Applied Signal Processing*, vol. 2002, no. 6, pp. 613–621, March 2002.

[4] S. Ye, E.C. Chang, and Q. Sun, "Watermarking based image authentication using feature amplification," in *International Conference on Multimedia and Expo (ICME)*, 2005, pp. 610–613.

[5] T.T. Ng, S.F. Chang, C.Y. Lin, and Q.Sun, "Passive-blind image forensics," in *Multimedia Security Technologies for Digital Rights*, chapter 06. Elsvier, 2006.

[6] M.K. Johnson and H. Farid, "Exposing digital forgeries by detecting inconsistencies in lighting," in *ACM Multimedia and Security Workshop*, 2005, pp. 1–10.

[7] Y.F. Hsu and S.F. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *IEEE International Conference on Multimedia and Expo*, July 2006, pp. 549–552.

[8] Y.Q. Shi, C. Chen, and G. Xuan, "Steganalysis versus splicing detection," in *International Workshop on Digital Watermarking*, December 2007.

[9] W. Chen, Y.Q. Shi, and G. Xuan, "Identifying computer graphics using hsv color model and statistical moments of characteristic functions," in *2007 IEEE International Conference on Multimedia and Expo*, July 2007, pp. 1123–1126.

[10] T.T. Ng, S.F. Chang, and Q. Sun, "A data set of authentic and spliced image blocks," Tech. Rep., DVMM, Columbia University, 2004, Dataset: http://www.ee.columbia.edu/dvmm/researchProjects/AuthenticationWatermarking/BlindImageVideoForensic/.

[11] D. Zou, Y.Q. Shi, W. Su, and G.R. Xuan, "Steganalysis based on markov model of tresholded prediction-error image," in *International Conference on Multimedia and Expo*, 2006.

[12] K. Tieu and P. Viola, "Boosting Image Retrieval," *International Journal of Computer Vision*, vol. 56, no. 1, pp. 17–36, 2004.

[13] J. Dong, X. Chen, L. Guo, and T. Tan, "Fusion based blind image steganalysis by boosting feature," in *Digital Watermarking*, vol. 5041, pp. 87–98. Springer, 2008.

[14] C.C. Chang and C.J. Lin, *LIBSVM: a library for support vector machines*, 2001, http://www.csie.ntu.edu.tw/∼cjlin/libsvm.