Forgery Localization Based on Image Chroma Feature Extraction

Areej S. Alfraih*, Johann A. Briffa, and Stephan Wesemeyer

*Dept. of Computing, University of Surrey, Guildford GU2 7XH, UK. Email: a.alfraih@surrey.ac.uk

Keywords: passive forensics, tamper detection techniques, GLCM, forgery localization, feature extraction.

Abstract

Many passive image tamper detection techniques have been presented in the expanding field of image forensics. Some of these techniques use a classifier for a final decision based on whole image statistics, resulting in a lack of forgery localization. The aim of this paper is to add localization to a previously published algorithm that uses grey-level co-occurrence matrix (GLCM) for extracting texture features from the chromatic component of an image (Cb or Cr component). Experimental results show that we can localize tampering for different sized regions with reasonable accuracy. The main trade-off is a diminishing detection accuracy as the region size decreases.

1 Introduction

Multimedia validity is becoming a major issue of concern nowadays due to the ease with which one can modify media using readily available software. As a result, the field of image forensics is targeted towards studying and analyzing multimedia to confirm authenticity or tampering. Image forensic tools can be classified into two main categories: active and passive. Watermarking, for example, is a well-known active image forensic tool where data is embedded into an image during the acquisition process. On the other hand, passive forensic tools do not depend on any prior data at all. Therefore, the analysis is performed on a blind basis.

Many algorithms based on feature extraction have been presented. Most of these use either grey scale images or the luminance component of RBG color images. The number of techniques that rely on extracting features from chromatic component of images is far smaller. We start by reviewing techniques that extract features from grey scale or color images first, followed by techniques that extract features from the chrominance component.

Davarzani et al. [6] present a technique that relies on block feature extraction using Multiresolution Local Binary Patterns (MLBP) for copy-move forgery detection. Their technique can efficiently detect duplicated regions even if they were rotated, scaled, blurred or compressed. Similarly, Amerini et al. [3] use feature extraction for copy-move attack detection. This method uses Scale-invariant feature transform (or SIFT) to detect duplicate regions. Dong et al. [8] present a technique that relies on the concept of pixel "run" which gives the number of consecutive pixels having the same gray level intensity with respect to a particular linear alignment. Although the method produces the desired results its accuracy level ranges between 69.75% and 84.36% depending on feature sets used.

In another work, Shi et al. [16] present a method that extracts Markov transition probabilities from the test image. The model works effectively on the Columbia Image Splicing Detection Evaluation Dataset [1]. Experimental results show that tampering can be detected with 92% accuracy. Liu et al. present a splicing detection algorithm that is based on image edge analysis and blur detection [10]. The Blurring operation averages the values of neighboring pixels in order to give a smooth visual effect. Therefore, the algorithm is designed to analyze the blur features that were introduced to the image then detect the changes in pixel values. The main drawback is that the algorithm is specific to grey-scale images and therefore a grey-scale conversion operation has to be performed prior to testing color images.

Carvalho et al. [7] present a method that extracts texture and edge based features from color images. Classification is then performed using an SVM classifier. Pan and Lyu [14] extract SIFT features then determine duplicate regions based on a feature matching technique. Similarly, Chen et al. [5] present a method that detects Harris corner interest points in an image, then statistical analysis is performed to represent image regions around Harris points.

Ghulam Muhammad [11] extract features from the chrominance component of an image for image tamper detection. The method computes the Weber Pattern (WP) histogram which is used as a texture feature for the image. A classifier is then used to make a final decision on the image. This method shows an increase of 18% in detection accuracy compared to the method by Peng et al. [15] which extracts compound statistical features from grey scale images for tamper detection. Hussain et al. [9] present a method that extracts Weber Law Descriptors (WLD) from the chrominance component of images. An SVM classifier is then used to make a final decision on the whole image. Wang et al. [18] present a technique that analyzes inconsistencies at the pixel level to detect image forgery. The algorithm detects splicing in images based on extracting texture information from image chroma. The interesting aspect about this technique is that it uses the chroma components of an image for tamper detection. The technique is robust and yields very good results, and it is one of the few techniques in the literature, as we mentioned previously, that uses the chrominance component of an image for tamper detection.

Experimental results of the last three techniques show that the chrominace component of images is very useful and can outperform results taken from grey scale or luminance component of RGB color images. For this reason, Wang et al.'s technique [18] was chosen for conducting further research.

This paper is organized as follows: section 2 briefly explains the published technique. Section 3 gives details about the conducted experiment. Section 4 shows experimental results. Finally, Section 5 concludes.

2 Background

The first step in the algorithm of [18] is to separate the image into its Y, Cb, and Cr components. The YCbCr is a color space used for digital images. The Y component represents the luminance, the Cb represents the blue difference chroma component and Cr represents the red difference chroma component. Most of image content is preserved in the Y component. Therefore, the Cb and Cr components don't show as much image content as the Y component. However, according to [18], the splicing process leaves traces that are more visible in the Cb or Cr components than the Y component. For example, spliced regions will have sharp edges while the authentic objects in the image will have smooth edges. An edge detector was used on the chroma component of the image before applying feature extraction. Wang et al. adopted a simple detector that generates edge images E_h, E_v, E_d, E_{-d} (Equations 1-4) as follows [12]:

$$E_h(i,j) = |x(i+1,j) - x(i,j)| \quad (1)$$

$$E_v(i,j) = |x(i,j+1) - x(i,j)| \quad (2)$$

$$E_d(i,j) = |x(i+1,j+1) - x(i,j)| \quad (3)$$

$$E\_d(i,j) = |x(i+1,j-1) - x(i,j)| \quad (4)$$

where $x(i,j)$ represents the gray value of a pixel at row i column j.

The grey-level co-occurrence matrix (GLCM) is used to extract second-order texture information from the image. GLCM computes the joint probability distribution function (PDF) of grey-level pairs in an image. The aim of feature extraction is to select certain texture characteristics from an image to be able to group them in clusters. Classification can then be used to distinguish between certain textural features in the image.
Boosting feature selection (BFS) [17] is then applied to the extracted image features to obtain optimal features. A LibSVM classifier was then used for forgery detection.

3 Experimental Set-up

Our experiment involved dividing images into non overlapping blocks of the following sizes: 128×128, 64×64, 32×32, 16×16 and 8×8 pixels. Obviously, only forged blocks were picked from the forged images since we already have enough original blocks taken from the original images. A random selection process was performed on the blocks to be chosen for training and testing the classifier. Each random block was then split into its Y, Cb, and Cr components. Feature extraction was performed on the Cb component of each image block.

Images were downloaded from the Image Tampering Database Of Cloning With Modern Tools [13]. This database has 6 different sets of uncompressed images of size 1024×1024 pixels at 8 bpc. Each set has a different type of forgery attack performed with GIMP or Photoshop. The content-aware attack works by filling a certain region in an image with new data automatically generated based on the region's neighborhood. Clone stamp attack fills a region in an image with existing pixels from another region that is chosen manually. Copy-paste attack works by copying the the pixels of a region then pasting them as a new layer on top of existing pixels in another region. Table 1 lists all the datasets.

Set	Algorithm	Software	# of Images
A	Clone Stamp	Photoshop	150
B	Clone Stamp	GIMP	150
C	Content-aware	Photoshop	150
D	Content-aware	GIMP	150
E	Copy-paste	Photoshop	75
F	Copy-paste	GIMP	75

Table 1. Datasets Used in Experiment

A Support Vector Machine [4] was used for testing image blocks. At each size, blocks were divided separately into a training set and a testing set. Half of the training set was authentic and the other half was tampered. The testing set contained 100 authentic and 100 tampered blocks. Training was performed once for each block size independently for each of the six tampered image sets. Tampered blocks were determined by comparing individual blocks from the original image to the corresponding blocks from the tampered image. When there is a change in any pixel value, the blocks is considered tampered. An RBF kernel was chosen to generate the model to be used for prediction/testing. Cross validation and grid search was performed on the training set to obtain the optimal RBF kernel parameters. It was important to have an equal amount of original and forged blocks in the training set and testing set (although the total number of blocks in the training sets may vary for different block sizes while all testing sets have 200 blocks total). The reason behind this is to ensure we have the same confidence for false positive and false negative results and not just the overall accuracy. The binomial proportion confidence interval for the final accuracy was computed as follows (Equation 5):

$$\hat{p} \pm z_{1-\frac{1}{2}\alpha}\sqrt{\frac{1}{n}\hat{p}(1-\hat{p})}, \quad (5)$$

where $\hat{p}$, n and $z_{1-\frac{1}{2}\alpha}$ are the estimated classification accuracy, testing sample size and confidence factor respectively.

4 Results

The results show that the block size plays a role in the detection accuracy. The smaller the block size the lower the detection accuracy. There are some cases where smaller blocks achieved a slightly better accuracy than larger blocks, however, the accuracy results still remains within each others' confidence inter-

val. Table 2 shows the obtained results for the different block sizes for each set along with the confidence interval for each result. All confidence intervals were computed for 95% confidence. The results were similar to what we expected, the accuracy decreases for smaller block sizes because the algorithm works by detecting 'sharp edges' of tampered regions. Therefore, when a block does not contain any pixels from the edges of the forged region it will not be detected as forged.

In order to validate this claim an experiment was made where 50 tampered blocks were manually selected. Half of the blocks contained tampered edges (i.e. the blocks contained a mixture of original and tampered pixels). The other half of tampered blocks did not contain any tampered edges (i.e. all the pixels in the blocks were tampered). The classifier was then used for testing each category of blocks separately. There was an increase of 8% in detection accuracy and a decrease of 8% in the false negative rate (FNR) when tampered edges were present in the blocks.

The ROC curves in Figure 1 show the TPR and FPR for each block size in each set. It can be seen from the graph that the type of tampering attack affects the detection results. This may be caused by the extracted features being sensitive to certain tampering attacks more than others. The best ROC performance was acheived by the 64×64 blocks in set D. The TPR was 88% and FPR 30%.

Our proposed technique improves on Wang et al. [18] by adding localization of tampered regions. We also tested on the CASIA TIDE database [2] so we can compare our results with two recent state-of-the-art techniques that used the same database. We chose the first three block sizes in our experiment (128×128, 64×64 and 32×32) since they generated the best results in the previous experiment. We trained and tested the SVM classifier for each block size separately. Then we reported the highest detection accuracy which was for the 32×32 blocks. The techniques we used for comparison were by Chen et al. [5] and Pan and Lyu [14]. Both techniques are robust and yield very good results. However, they do not show the efficacy of their technique in detecting attacks performed by different algorithms. We used the detection accuracy along with the false positive rate (FPR) and false negative rate (FNR) as metrics for comparison with these techniques. The detection accuracy refers to the proportion of true positives (tampered detected as tampered) and true negatives (original detected as original). FPR is the proportion of original blocks that were falsely detected as tampered. FNR is the proportion of tampered blocks that were falsely detected as original. Our technique achieves a slightly better detection accuracy than the other two techniques (93%). However, the FPR is a little higher than the other two techniques (4%) probably because our technique is sensitive to the presence/absence of tampered edges within the blocks as we previously mentioned. Table 3 shows a comparison between our localization technique and the other two techniques.

Technique	Accuracy%	FPR%	FNR%
Pan and Lyu [14]	89.96	1.25	18.84
Chen et al. [5]	92.15	3.30	12.40
Ours	93.0	4.0	10.0

Table 3. Comparison Between Techniques Based on CASIA TIDE Database [2]

5 Conclusion

The objective of this experiment was to determine whether or not we can localize forged regions in an image using an SVM classifier. Results show that this is possible but, the detection accuracy decreases as the block size decreases. The algorithm performs efficiently when the edge of a forged region is present within the block to be tested. The extracted features seemed to be more sensitive to certain types of forgery attacks and therefore, performed better in detecting them. Our technique also compares favourably with other state of the art techniques that implemented localization. In further work, we intend to investigate how localization may be affected by adding additional features in order to improve detection results.

Acknowledgements

A very special thank you goes to Dr. Wei Wang (Institute of Automation, Chinese Academy of Sciences (CASIA)) for his kind cooperation and feedback.

References

[1] Columbia image splicing detection evaluation dataset http://www.ee.columbia.edu/ln/dvmm/downloads/ authspliceddataset/dlform.html, 2004.

[2] Casia tampered image detection evaluation database. availabe at: http://forensics.idealtest.org, 2009.

[3] AMERINI, I., BALLAN, L., CALDELLI, R., BIMBO, A. D., AND SERRA, G. A sift-based forensic method for copy-move attack detection and transformation recovery. In *IEEE Transactions on Information Forensics and Security* (2011).

[4] CHANG, C.-C., AND LIN, C.-J. Libsvm : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* (2011).

[5] CHEN, L., LU, W., NI, J., SUN, W., AND HUANG, J. Region duplication detection based on harris corner points and step sector statistics. *Journal of Visual Communication & Image Representation 24* (2013), 244–254.

[6] DAVARZANI, R., YAGHMAIE, K., MOZAFFARI, S., AND TAPAK, M. Copy-move forgery detection using multiresolution local binary patterns. *Forensic Science International 231* (2013), 61–72.

Block Size	Set A	Set B	Set C	Set D	Set E	Set F
128×128	78.5% ±5.69%	59.5% ±6.80%	50% ±6.93%	63% ±6.69%	58.5% ±6.83%	56% ±6.88%
64×64	60.5% ±6.78%	50.5% ±6.93%	50% ±6.93%	69% ±6.41%	60% ±6.79%	58.5% ±6.83%
32×32	59.5% ±6.80%	50.5% ±6.93%	56.5% ±6.87%	56% ±6.88%	48.5 % ±6.93%	65.5% ±6.59%
16×16	62% ±6.73%	55.5% ±6.89%	55% ±6.89%	54.5% ±6.90%	52% ±6.92%	56.5% ±6.87%
8×8	53.5% ±6.91%	51.5% ±6.93%	53.5% ±6.91%	52.5% ±6.92%	51% ±6.93%	53% ±6.92%

Table 2. Detection Accuracy for Image Blocks from Different Sets

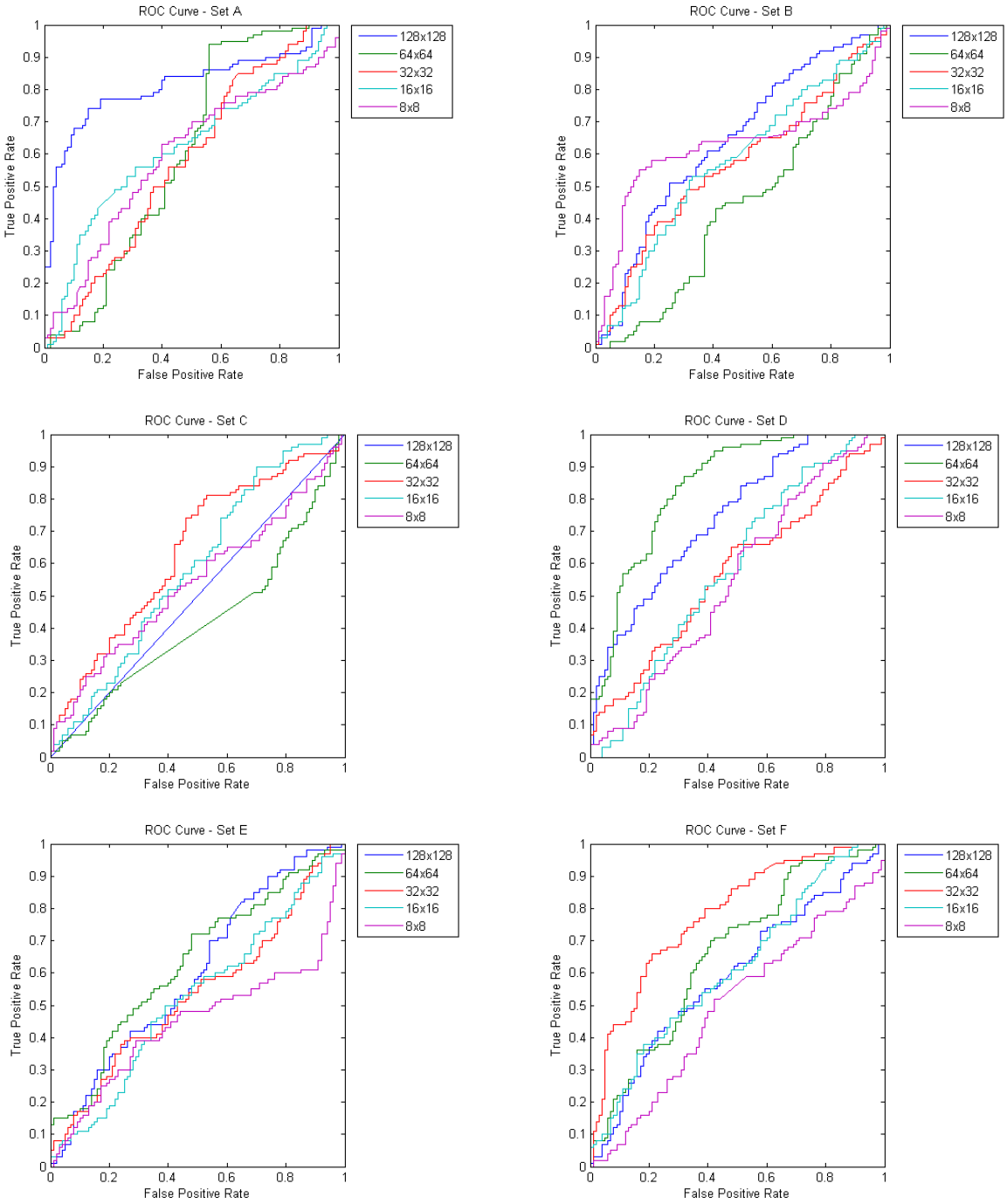

Figure 1. ROC Curves for Different Block Sizes in Different Sets

[7] DE CARVALHO, T. J., RIESS, C., ANGELOPOULOU, E., PEDRINI, H., AND DE REZENDE ROCHA, A. Exposing digital image forgries by illumination color classification. *IEEE Transactions on Information Forensics and Security (7) 8* (2013), 1182–1194.

[8] DONG, J., WANG, W., TAN, T., AND SHI, Y. Run-length and edge statistics based approach for image splicing detection. *Digital Watermarking* (2009), 76–87.

[9] HUSSAIN, M., MUHAMMAD, G., SALEH, S. Q., MIRZA, A. M., AND BEBIS, G. Copy-move image forgery detection using multi-resolution weber descriptors. In *Eighth International Conference on Signal Image Technology and Internet Based Systems* (2012).

[10] LIU, G., WANG, J., LIAN, S., AND DAI, Y. Detect image splicing with artificial blurred boundary. *Mathematical and Computer Modelling* (2011).

[11] MUHAMMAD, G. Multi-scale local texture descriptor for image forgery detection. In *IEEE International Conference on Industrial Technology (ICIT)* (2013).

[12] NG, T. T., CHANG, S. F., AND SUN, Q. Blind detection of photomontage using higher order statistics. In *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on* (2004), vol. 5, IEEE, pp. V–688–V–691 Vol. 5.

[13] NICOLAOU, D., AND BRIFFA, J. A. Image tampering database of cloning with modern tools (unpublished).

[14] PAN, X., AND LYU, S. Region duplication detection using image feature matching. *IEEE Transactions on Information Forensics and Security (4) 5* (2010), 857–867.

[15] PENG, F., NIE, Y., AND LONG, M. A complete passive blind image copy-move forensics scheme based on compound statistics features. *Forensic Science International 212(1-3)* (2011), e21–e25.

[16] SHI, Y. Q., CHEN, C., AND CHEN, W. A natural image model approach to splicing detection. In *Proceedings of the 9th workshop on Multimedia & security* (2007), ACM, pp. 51–62.

[17] TIEU, K., AND VIOLA, P. Boosting image retrieval. *International Journal of Computer Vision 1* (International Journal of Computer Vision), 228 – 235.

[18] WANG, W., DONG, J., AND TAN, T. Effective image splicing detection based on image chroma. In *Image Processing (ICIP), 2009 16th IEEE International Conference on* (2009), IEEE, pp. 1257–1260.