SOURCE SCANNER IDENTIFICATION FOR SCANNED DOCUMENTS

Nitin Khanna and Edward J. Delp

Video and Image Processing Laboratory
School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana USA

ABSTRACT

Recently there has been a great deal of interest using features intrinsic to a data-generating sensor for the purpose of source identification. Numerous methods have been proposed for different problems related to digital image forensics. The goal of our work is to identify the scanner used for generating a scanned (digital) version of a printed (hard-copy) document. In this paper we describe the use of texture analysis to identify the scanner used to scan a text document. The efficacy of our proposed method is also demonstrated.

Index Terms— scanner forensics, document forensics

1. INTRODUCTION

There are various levels at which the image source/sensor identification problem can be addressed [1]. One may want to find the particular device (digital camera or scanner) which generated the image or one might be interested in knowing only the make and model of the device. As summarized in [2, 3], a number of robust methods have been proposed for source camera identification. In [4], techniques for classification of images based on their sources: scanner, camera and computer generated images, are presented.

Sensor pattern noise can be successfully used for source camera identification and forgery detection [2, 5, 6]. Also, source scanner identification for photographs can be done using statistical features of sensor pattern noise [7, 8] . All these methods for source scanner identification focused on scanned versions of photographs and not on scanned versions of printed text documents. Since the methods utilizing sensor pattern noise for source identification mainly use Photo-Response Non-uniformity (PRNU) as the sensor's signature and the PRNU is almost absent in "saturated" regions of an image [5]. Therefore, these methods may not work for scanned documents which generally lack presence of continuous tones and are dominated by "saturated" pixels. In this paper we present methods for authenticating scanned text documents, that have been captured by flatbed desktop scanners, using texture features.

[1]This material is based upon work supported by the National Science Foundation under Grant No. CNS-0524540. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Address all correspondence to E. J. Delp at ace@ecn.purdue.edu.

2. SYSTEM OVERVIEW

Figure 1 shows the block diagram of our scanner identification system. Given a digital image of a text document scanned with an unknown source, henceforth referred to as the *unknown scanned document*, we want to be able to identify the scanner that created it.

Given an unknown scanned document, the first step is to extract all the letter "e"s in the document. The letter "e" is the most frequently occurring character in the English language. A set of features are extracted from each group of n_e characters ("e's") forming a feature vector for each group of n_e "e"s in the document. Further block level features are obtained by dividing the unknown scanned document into non-overlapping blocks of size $N_b \times N_b$. A different set of features are extracted from each of these blocks. Each of these feature vectors are then classified independently using different classifiers. The classifier used is a combination of Linear Discriminant Analysis (LDA) for dimensionality reduction and Support Vector Machine (SVM) for final class labeling. Let Ψ be the set of all scanners $\{S_1, S_2, \cdots, S_n\}$ (in our work this is the set of 5 scanners shown in Table 1). For any $\phi \epsilon \Psi$, let $c(\phi)$ be the number of groups of "e"s and scanned blocks classified as being scanned by scanner ϕ. The final classification is done by choosing ϕ such that $c(\phi)$ is maximum. In other words, a majority vote is performed on the resulting classifications from the SVM classifier

3. GRAYLEVEL CO-OCCURRENCE FEATURES

In contrast to scanned images, scanned documents generally lack presence of continuous tones and are dominated by "saturated" pixels. That is, most of the pixel values are either close to zero or are close to 255. This makes it very difficult to accurately use the type of signatures described in ear-

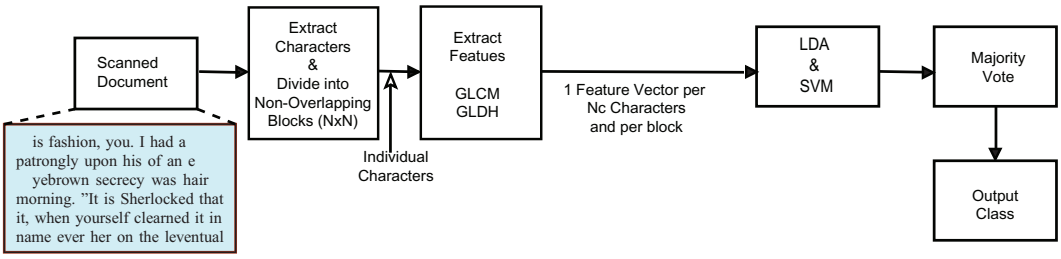


Fig. 1. System Diagram of the Proposed Scanner Identification System.

lier source camera forensics [5] or for scanner identification from images [8]. For example, pattern noise (such as Photo-Response Non-uniformity, PRNU) can not be used due to it's absence in "saturated" image regions [5]. Thus, a different set of features is needed to describe each scanner uniquely by observing an example of the output scanned documents. We will exploit the observation that depending upon the quality of the scanner, (i.e., it's sensitivity to sudden changes in gray-levels), the quality of edges in scanned documents will vary. More specifically, for a higher quality scanner characters will be represented by more solid black lines and the transition from black to white will be sharper. On the other hand, for a lower quality scanner, the black lines representing the characters will have more variations within them from black to higher gray levels and the transitions from black to white pixels will also be more gradual. This will result in changes in the texture features. These differences are quantified by extracting features from individual scanned characters, in particular "e"s. For documents scanned at low resolution such as 200dpi (which is generally the case for normal office usage), each character is very small, about 15×20 pixels and is non-convex, so it is difficult to filter the image in either the pixel or transform domain if we are interested only in the printed region of each character. The graylevel fluctuation in the scanned characters in the process direction can be modeled as textures [9]. We used graylevel co-occurrence texture features as described in [9] as well as two pixel based features. We have shown that these types of features can be very robust for identifying printed documents [9]. Further, to alleviate problems due to the very small size of individual characters and gather sufficient statistics to estimate the Gray-Level Co-occurrence Matrix (GLCM), these matrices are generated from a group of n_e "e"s at a time. In the experiments presented here, n_e was chosen to be 100.

Graylevel co-occurrence texture features assume that the texture information in an image is contained in the overall spatial relationships among the pixels in the image [9]. This is done by first determining the Graylevel Co-occurrence Matrix (GLCM), which is an estimate of the second order probability density function of the pixels in the image. The features are then the statistics obtained from the GLCM.

We assume that the texture in a document is predominantly in the process direction (that is, scan direction) as the same linear sensor is translated horizontally by a mechanical system to generate the complete scan. Figure 2 shows an idealized character, $Img(i,j)$, from which features are extracted. The region of interest (ROI) is the set of all pixels within the rectangular bounding box around the character. The determination of these bounding boxes is done by using the open source OCR system Ocrad [10].

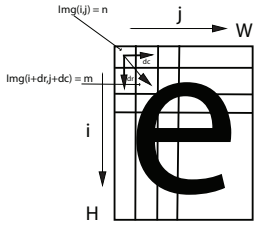


Fig. 2. Idealized Character for generation of $glcm(n,m)$.

The Gray-Level Co-occurrence Matrix (GLCM) is then estimated. This matrix, defined in Equation 1, has entries $glcm(n,m,dr,dc)$ which are equal to the number of occurrences of pixels with graylevels n and m respectively with a separation of (dr,dc) pixels (Figure 2). If the GLCM is normalized such that its entries sum to one, the entries then represent the probability of occurrence of pixel pairs with graylevels n and m with separation (dr,dc). For generating features from each character (character level features), we will choose $dc = 0$ and $dr = 1$.

$$glcm(n,m,dr,dc) = \sum_{(i,j),(i+dr,j+dc)\epsilon ROI} 1_{\{Img(i,j)=n, Img(i+dr,j+dc)=m\}} \quad (1)$$

These GLCM metrics are estimated for each of the extracted "e"s and average GLCM is obtained for each group of n_e "e"s. The twenty two statistical features extracted from each of these average GLCMs are as described in [9] and are not repeated here due to space limitations. In contrast to printer identification application [9], due to very small of size characters for 200 dpi scans, using features from GLCM's corresponding to each character separately does not give good classification results as demonstrated by our initial experiments.

These twenty two features from anisotropic GLCM (corresponding to dr =1, and dc =0) are extracted from each group of n_e "e"s and separately from each non-overlapping block of $N_b \times N_b$ pixels. Further, for each block of $N_b \times N_b$ pixels, an isotropic gray-level difference histogram (GLDH) is also used as another 246 dimensional signature. The isotropic GLDH with d =1 is defined in Equations 2 and 3 (where $glcm(n, m, dr, dc)$ is in Equation 1). Note that in defining the isotropic GLDH, lower values of k are not used and so range of k is $[10, 255]$. These lower values of k will correspond to completely black or completely white regions and so are not useful as scanner signature and will also vary from block to block depending upon what percentage of block's area corresponds to the background. The isotropic GLDH defined in Equation 3 is normalized to have sum equal to one before using it as scanner signature.

$$glcm_{isotropic}(n, m) = \sum_{dr=-1}^{1} \sum_{\substack{dc=-1 \\ (dr,dc)\neq(0,0)}}^{1} glcm(n, m, dr, dc) \tag{2}$$

$$gldh_{isotropic}(k) = \sum_{\substack{0\leq n\leq 255 \\ 0\leq m\leq 255 \\ |n-m|=k}} glcm_{isotropic}(n, m), \;\; k \in [0, 255] \tag{3}$$

Hence, corresponding to an unknown scanned document with N_e "e"s and size $N \times M$ pixels, we get around N_e/n_e 22-dimensional features for each group of "e"s, around $(N \times M)/(N_b \times N_b)$ 22-dimensional features for each block of size $N_b \times N_b$ pixels and around $(N \times M)/(N_b \times N_b)$ 246-dimensional features for each block of size $N_b \times N_b$ pixels. The final decision about source scanner is taken as majority voting over these $(N_e/n_e + 2 * (N \times M)/(N_b \times N_b))$ individual decisions.

4. EXPERIMENTAL RESULTS

The classifiers must be trained, for generating test and train datasets, the Forensic Monkey Text Generator (FMTG) (as described in [11]) is used to create random documents with known statistics. Using the FMTG, it is estimated that in a page of English text printed at 10 point font there are on average 630 "e"s per page [11]. For our experiments, 25 test documents (generated using FMTG, at 10 point, Times New Roman font) are printed with a consumer quality laser printer (HP Laserjet 3800dn). All the documents are printed on similar quality paper and using the same printer to make sure that we are addressing the variability due to the scanners rather than the variation in paper quality or printer. The 25 test documents are scanned at 200dpi using each of the five scanners shown in Table 1. To meet the requirements of most common usage, the pages are scanned at low resolution (200 dpi) with 8 bits/pixel (grayscale). In the experiments, n_e is chosen to be 100 and N_b is chosen to be 512. Thus for each of these documents of A4 size, scanned at 200 dpi, there are around 6 character level feature vectors and around 2×12 block level feature vectors.

Three separate classifiers (LDA + SVM) are trained for each class of features, namely GLCM features from groups of "e"s, GLCM features from each of the blocks of size $N_b \times N_b$ and isotropic GLDH features from each of the blocks of size $N_b \times N_b$. The character level classifier (using 22-dimensional feature vector from each group of n_e "e"s) is trained with randomly chosen 375 known feature vectors and tested over a different set of 375 feature vectors. The training and testing sets are made up of 75 feature vectors from each of 5 scanners listed in Table 1. Two block level classifiers (one using 22-dimensional GLCM feature and another using 246 dimensional isotropic GLDH) are trained with randomly chosen 750 known feature vectors and tested over a different set of 750 feature vectors. The training and testing sets are made up of 150 feature vectors from each of 5 scanners listed in Table 1. Each of these feature vectors are independent of one another. The classifier training and testing phases are repeated 100 times to obtain the final performance measures.

Table 1. Scanners Used For Classification.

	Make/Model	Sensor	Native Resolution
S_1	Epson 4490 Photo	CCD	4800 DPI
S_2	OpticSlim 2420	CIS	1200 DPI
S_3	Canon LiDE 25	CIS	1200 DPI
S_4	Canon LiDE 70	CIS	1200 DPI
S_5	Canon LiDE 100	CIS	2400 DPI

Figure 3 shows portions of the sample images scanned with different scanners. It is easily seen that in some cases these images are visually differentiable due to changes in brightness and contrast settings. Since an unknown document might not be scanned at default scanner settings and the used brightness and contrast settings might be unknown. Therefore, before source scanner identification, the images are post-processed to be visually more similar by adjusting the parameters of a linear intensity transform. This will help to ensure that the proposed system will work even when the documents are scanned with different brightness and contrast settings or latter post-processed by linear intensity transformations.

To demonstrate the efficacy of proposed features in source scanner identification, we plotted two dimensional scatter plots showing the separability of these five scanner classes in low dimensional feature space. Figure 4 shows the scatter plot for two manually chosen character level features of scanned images saved in TIF format. Even though all the classes do not separate completely, the two features still have good discrimination capability. The efficacy is further demonstrated after using Linear Discriminant Analysis (LDA) on 22-dimensional character level features. These are

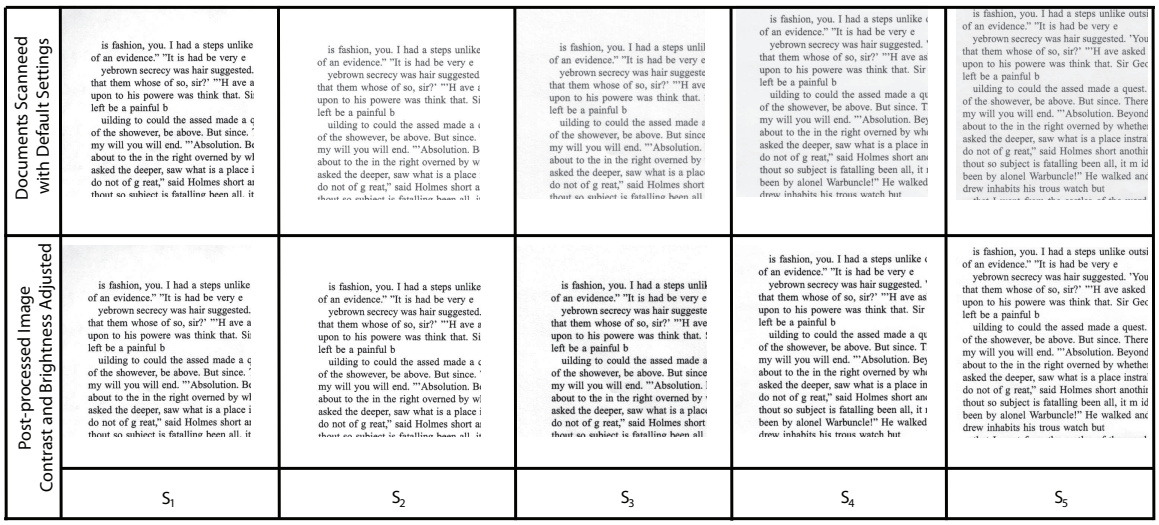


Fig. 3. Portions of Sample Images from Different Scanners.

projected into a 7-dimensional feature space. Figure 5 shows scatter plots for the two projected features with maximum discrimination.

Table 2 shows average accuracy of the dedicated classifiers for scanned documents saved in different formats. The classifiers are trained and tested on feature vectors coming from scanned documents saved in the same format. Note that the accuracies are for classifying individual feature vectors and not the complete document. In all these cases, the accuracy for classifying complete documents is 100%. To see the effectiveness of the proposed scheme in scenarios where the JPEG quality factor may not be reliablly known or estimated, another set of three general classifiers are trained and tested on randomly chosen feature vectors from images saved with two different JPEG quality factors (Q =80, and 60). Both training and testing sets include features from documents saved at different quality factors. Table 3 shows the confusion matrix for a block level isotropic GLDH features which has average classification accuracy of 98%. Similar general classifiers for the character level GLCM statistics and block level GLCM statistics have average accuracies 99.7% and 99.4% respectively. In all our experiments, after majority voting the source scanner amongst five scanners is found with 100% classification accuracy.

Table 2. Average Accuracy of Dedicated Classifier.

Image Format	Feature Type	Average Accuracy
TIF	Character Level GLCM	99.9
	Block Level GLCM	99.9
	Block Level GLDH	96.4
JPEG (Q =80)	Character Level GLCM	99.7
	Block Level GLCM	99.7
	Block Level GLDH	98
JPEG (Q =60)	Character Level GLCM	99.6
	Block Level GLCM	99.5
	Block Level GLDH	95.2

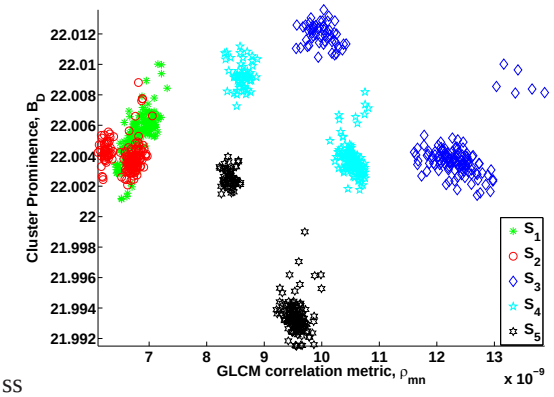
5. CONCLUSION

This paper proposed methods for source scanner identification for scanned text documents using texture features. As shown by the experiments, the proposed method is robust to JPEG compression and gives 100% classification accuracy for classifying A4 size text documents and more than 95% classification accuracy for classifying smaller blocks of size 512×512. The proposed features are also robust to the scan area used for a particular scan, that is we do not need to know which exact location of document on scanner's bed was used for scanning. Future work will include experiments to see the effect of variation in font sizes and fonts and tests on more scanners. As with every other method for image forensic, the proposed scheme also has limitations and will fail to work in certain circumstances such as heavy post-processing of scanned documents.

We thank our colleague Aravind Mikkilineni of the Video and Image Processing Laboratory at Purdue University for suggesting the use of the features we developed for printer analysis to this problem.

6. REFERENCES

[1] P.-J. Chiang, N. Khanna, A. K. Mikkilineni, M. V. O. Segovia, S. Suh, J. P. Allebach, G. T.-C. Chiu, and E. J. Delp, "Printer and scanner forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 72–83, March 2009.



SS

Fig. 4. Scatter Plot for Two Manually Chosen Character Level Features (giving best separation in 2-D feature space) of TIF Images.

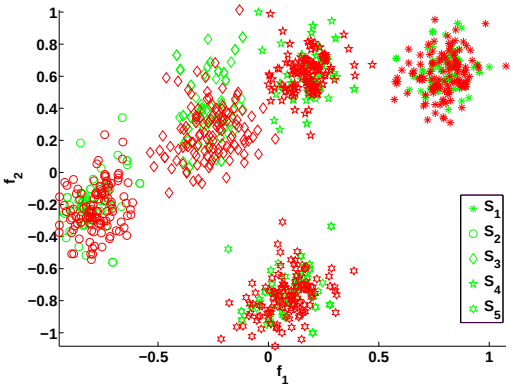


Fig. 5. Scatter Plot for Two Manually Chosen Character Level Features (after performing LDA) of TIF Images, (green symbols correspond to the feature vectors used for training LDA and red corresponds the feature vectors used for testing).

Table 3. Confusion Matrix for General Classifier (testing and training on JPEG images with Q =80 and 60)

		Predicted				
		S_1	S_2	S_3	S_4	S_5
Actual	S_1	96.9	2.9	0	0.0	0.2
	S_2	3.4	96.6	0	0	0.0
	S_3	0	0	98.2	1.0	0.8
	S_4	0	0	0.4	99.5	0.1
	S_5	0	0	0.3	0.2	99.5

[2] J. Fridrich, "Digital image forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 26–37, March 2009.

[3] N. Khanna, A. K. Mikkilineni, A. F. Martone, G. N. Ali, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "A survey of forensic characterization methods for physical devices," *Digital Investigation*, vol. 3, pp. 17–28, 2006.

[4] N. Khanna, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," *Proceedings of the 2008 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, March 2008, pp. 1653–1656.

[5] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, March 2008.

[6] N. Khanna, A. K. Mikkilineni, and E. J. Delp, "Forensic camera classification: Verification of sensor pattern noise approach," *Forensic Science Communications (FSC)*, vol. 11, no. 1, January 2009.

[7] H. Gou, A. Swaminathan, and M. Wu, "Robust scanner identification based on noise features," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1. SPIE, 2007, p. 65050S.

[8] N. Khanna, A. K. Mikkilineni, and E. J. Delp, "Scanner identification using feature-based processing and analysis," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 123–139, March 2009.

[9] A. K. Mikkilineni, P.-J. Chiang, G. N. Ali, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Printer identification based on textural features," *Proceedings of the IS&T's NIP20: International Conference on Digital Printing Technologies*, vol. 20, Salt Lake City, UT, October/November 2004, pp. 306–311.

[10] (2009) Ocrad - the gnu ocr. http://www.gnu.org/software/ocrad/

[11] A. K. Mikkilineni, G. N. Ali, P.-J. Chiang, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Signature-embedding in printed documents for security and forensic applications," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents VI*, vol. 5306, San Jose, CA, January 2004, pp. 455–466.