

# Texture Based Attacks on Intrinsic Signature Based Printer Identification

Aravind K. Mikkilineni, Nitin Khanna, Edward J. Delp  
School of Electrical and Computer Engineering  
Purdue University, West Lafayette, Indiana USA

## ABSTRACT

Several methods exist for printer identification from a printed document. We have developed a system that performs printer identification using intrinsic signatures of the printers. Because an intrinsic signature is tied directly to the electromechanical properties of the printer, it is difficult to forge or remove. There are many instances where existence of the intrinsic signature in the printed document is undesirable. In this work we explore texture based attacks on intrinsic printer identification from text documents. An updated intrinsic printer identification system is presented that merges both texture and banding features. It is shown that this system is scalable and robust against several types of attacks that one may use in an attempt to obscure the intrinsic signature.

**Keywords:** document security, secure printing, printer identification, banding

## 1. INTRODUCTION

In today's digital world securing different forms of content is very important in terms of protecting copyright and verifying authenticity. One example is watermarking of digital audio and images. We believe that a marking scheme analogous to digital watermarking but for documents is very important. Printed material is a direct accessory to many criminal and terrorist acts. Examples include forgery or alteration of documents used for purposes of identity, security, or recording transactions. In addition, printed material may be used in the course of conducting illicit or terrorist activities. Examples include instruction manuals, team rosters, meeting notes, and correspondence. In both cases, the ability to identify the device or type of device used to print the material in question would provide a valuable aid for law enforcement and intelligence agencies. We also believe that average users need to be able to print secure documents, for example boarding passes and bank transactions.

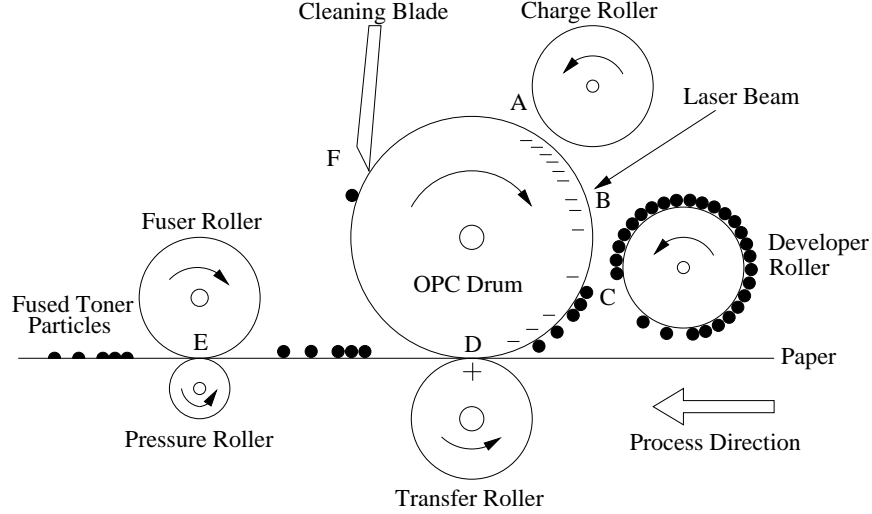
There currently exist techniques to secure documents such as bank notes using paper watermarks, security fibers, holograms, or special inks. The problem is that the use of these security techniques can be cost prohibitive. Most of these techniques either require special equipment to embed the security features, or are simply too expensive for an average consumer. Additionally, there are a number of applications in which it is desirable to be able to identify the technology, manufacturer, model, or even specific unit that was used to print a given document. At the same time, there may also exist a need for some level of anonymity when distributing a printed document.

Various methods for identifying the source of printed documents have been developed. Most of these methods rely on modifying the document before it is printed to embed information such as the printer serial number. We have taken a different approach to printer identification. Methods previously proposed by us have focused on information embedding by modification of the print process. By moving the encoding to printer hardware we gain more control over the type of marks that can be added to the document and also deter hacking or modification of the encoded data before it is printed. This embedded information is referred to as the extrinsic signature of the document. Additionally, there are features present in the document that are characteristic of the specific type and model of printer which created it. These are referred to as intrinsic features and are used as an intrinsic signature of the specific printer.

Identification of the printer which produced a document is possible using the printer's intrinsic signature. There exist many electromechanical imperfections in a printer which cause print quality defects on the printed

---

Address all correspondence to E. J. Delp at [ace@ecn.purdue.edu](mailto:ace@ecn.purdue.edu)



**Figure 1.** Diagram of the EP process: (A) charging, (B) exposure, (C) development, (D) transfer, (E) fusing, (F) cleaning

page. The major artifact in the printed output is banding which appears as alternating light and dark bands perpendicular to the process direction (the direction the paper passes through the printer). The appearance of this artifact is directly related to (Optical Photo-Conducting) OPC drum imperfections, charge roller defects, and polygon mirror error due to wobble or imperfections. To identify a specific printer a frequency analysis of the banding present in the document can be performed and the resulting data classified using standard techniques.

More recently we have viewed the print quality defects as a texture. Texture features are used for intrinsic identification from printed text documents. Using a combination of graylevel co-occurrence and pixel based features we have been able to correctly classify documents with a high degree of accuracy in a 10 printer testbed.

There are many instances where existence of the intrinsic signature in the printed document is undesirable. This is useful, for example, in protecting the anonymity of people distributing printed documents during peaceful protest. On the other hand, some groups may want to hide the intrinsic signature for illegal purposes, such as distribution of counterfeit currency. In this work we explore texture based attacks on intrinsic printer identification from text documents. Understanding these types of attacks will lead to improvement of our intrinsic signature based identification methods.

The only way to achieve complete removal of the intrinsic signature is through extensive, non-trivial, and perhaps impossible hardware modifications of the printer. Instead of complete removal, we focus on attacking the graylevel co-occurrence matrix based intrinsic signature detection for text documents. The proposed method embeds false textures in each character of a text document. The changes in each character’s texture are performed before the data is sent to the printer, so no hardware modifications are necessary. The false texture is designed to prevent correct printer identification by changing the graylevel co-occurrence features used as intrinsic signatures of the printer such that they do not lead to correct printer identification. Quantitative measures of print quality are used to ensure that false textures in the character do not excessively degrade the print quality.

## 2. INTRINSIC SIGNATURES

Figure 1 shows a side view of a typical EP printer. The print process has six steps. The first step is to uniformly charge the optical photoconductor (OPC) drum. Next a laser scans the drum and discharges specific locations on the drum. The discharged locations on the drum attract toner particles which are then attracted to the paper which has an opposite charge. Next the paper with the toner particles on it passes through a fuser and pressure

roller which melt and permanently affix the toner to the paper. Finally a blade or brush cleans any excess toner from the OPC drum.

Laser printers can be characterized using intrinsic signatures such as banding[1]. Banding is an artifact caused by fluctuations of the OPC angular velocity and errors in the gear transmission mechanism. It appears as nonuniform light and dark lines perpendicular to the process direction. This is the direction in which the paper moves through the printer. Different printers have different sets of banding frequencies depending upon brand and model.

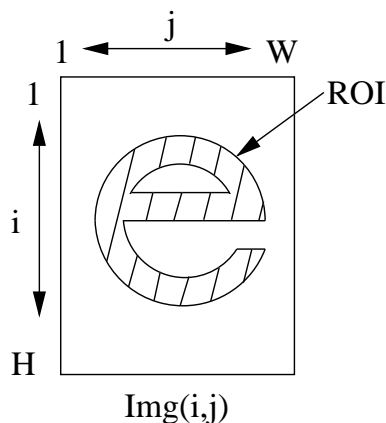
Banding-based identification is based on frequency domain analysis of a one-dimensional projected signal of mid-tone regions of the document, typically occurring in halftone printed images. Fourier analysis of the signal yields the banding frequencies.

In a text-only document, the absence of large midtone areas makes it difficult to capture suitable signals for banding analysis according to the method just described. In this case, texture features estimated from individual text characters, can be used to capture the intrinsic signature. Texture is a consequence of the fluctuations in the developed toner due to electromechanical imperfections. A set of texture features are based on the graylevel co-occurrence matrix (GLCM)[2, 3]. These features are estimated from printed text regions and are classified using pattern recognition techniques.

We want to be able to determine a set of features that can be used to describe each printer uniquely by observing an example of the output of the printer. We will treat the output scanned document as an “image” and use image analysis tools to determine the features that characterize the printer. We will accomplish this by extracting features from individual printed characters, in particular an “e”. Each character is very small, about 180x160 pixels and is non-convex, so it is difficult to perform any meaningful filtering operations in either the pixel or transform domain if we are interested only in the printed region of each character. The printed areas of the document have fluctuations which can be viewed as texture. To model the texture we used graylevel co-occurrence texture features as described in[4, 5] as well as two pixel based features.

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.[4] This is done by first determining the Graylevel Co-occurrence Matrix (GLCM). This is an estimate of the second order probability density function of the pixels in the image. The features are then statistics obtained from the GLCM.

We assume that the texture in a document is predominantly in the process direction[6]. 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 printed area of the character. The determination of this region involves morphological filtering and is discussed in[7].

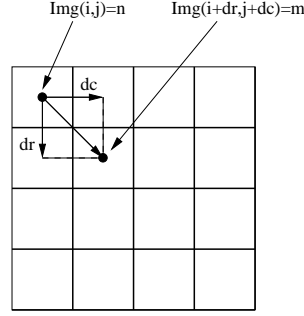


**Figure 2.** Idealized character

We define the number of pixels in the ROI to be

$$R = \sum_{(i,j) \in ROI} 1. \quad (1)$$

We then estimate the Gray-Level Co-occurrence Matrix (GLCM). This matrix, defined in Equation 2, has entries  $g_{lcm}(n, m)$  which are equal to the number of occurrences of pixels with graylevels  $n$  and  $m$  respectively with a separation of  $(dr, dc)$  pixels (see Figure 3). The number of pixels over which this estimate is obtained is given by Equation 3. If the GLCM is normalized with respect to  $R_{g_{lcm}}$ , its entries then represent the probability of occurrence of pixel pairs with graylevels  $n$  and  $m$  with separation  $(dr, dc)$ . We will choose  $dc = 0$  and  $dr = 1$ .



**Figure 3.** Generation of  $g_{lcm}(n, m)$

$$g_{lcm}(n, m) = \sum_{(i,j), (i+dr, j+dc) \in ROI} 1_{\{Img(i,j)=n, Img(i+dr, j+dc)=m\}} \quad (2)$$

$$R_{g_{lcm}} = \sum_{(i,j), (i+dr, j+dc) \in ROI} 1 \quad (3)$$

$$p_{g_{lcm}}(n, m) = \frac{1}{R_{g_{lcm}}} g_{lcm}(n, m) \quad (4)$$

Twenty-two features are obtained from the GLCM and are defined in [8]. These features are the marginal means and variances of the GLCM; energy; three entropy measures; maximum entry; two correlation metrics; energy, entropy, inertia, and local homogeneity of the difference histogram; and finally the energy, entropy, variance, cluster shade, and cluster prominence of the sum histogram.

Two pixel based features are also included. These are the variance and entropy of the pixel values in the ROI.

In addition to the GLCM and pixel features above, we add 15 banding features to the feature vector. This seems counter-productive considering that banding is difficult to estimate from a saturated region such as a character of text. However, the attacks we will consider may cause the printer to use halftoning when printing the text, in which case banding features can be more easily estimated.

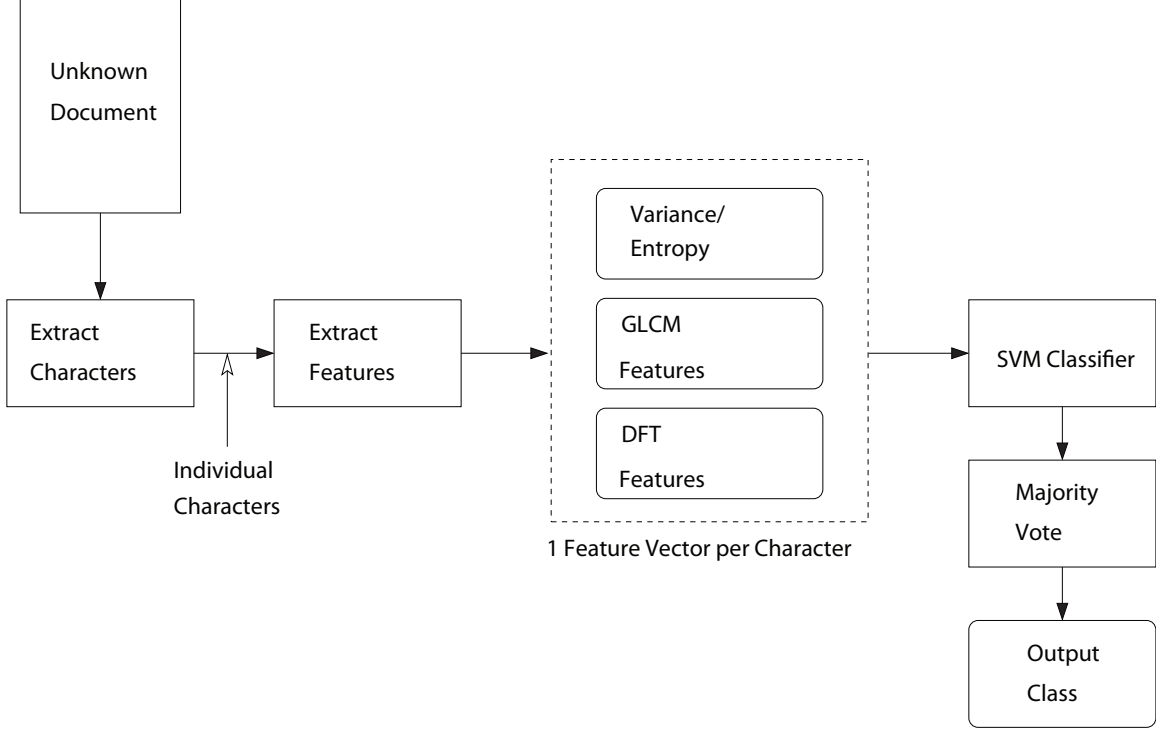
To obtain the banding features, a normalized projection of the character is first obtained as

$$b(i) = \frac{\sum_{(i,j) \in ROI} Img(i, j)}{\sum_{(i,j) \in ROI} 1}. \quad (5)$$

The banding features are taken as the magnitudes of the first 15 points of the DFT spectrum defined as

$$P_b(n) = \sum_{k=0}^{239} b(k) e^{-j \frac{2\pi kn}{240}}. \quad (6)$$

The first 15 points correspond to bins centered at  $\{10, 20, \dots, 150\}$  cycles/inch.



**Figure 4.** System diagram of printer identification scheme

**Table 1.** Printers used for classification.

Printer Identifier	Manufacturer	Model	DPI
$P_1$	HP	Color LaserJet 2605	1200
$P_2$	HP	Color LaserJet 3800	1200
$P_3$	Samsung	ML-1450	600
$P_4$	HP	LaserJet 4450	600

### 3. SYSTEM OVERVIEW

Figure 4 shows the block diagram of our intrinsic printer identification scheme using the features described in the previous section. Given a document with an unknown source, we want to be able to identify the printer that created it.

The first step is to scan the document at 2400 dpi with 8 bits/pixel (grayscale). Next all the letter “e”s in the document are extracted. Features are extracted from each character forming a feature vector for each letter “e” in the document. Each feature vector is then classified individually using a support vector machine (SVM) classifier[9]. The training set for the SVM classifier consists of 1500 feature vectors from each of 4 printers listed in Table 1. Each of these feature vectors are independent of one another.

Let  $\Psi$  be the set of all printers  $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$  (in our work these are the 4 printers shown in Table 1). For any  $\phi \in \Psi$ , let  $c(\phi)$  be the number of “e”s classified as being printed by printer  $\phi$ . The final classification is decided by choosing  $\phi$  such that  $c(\phi)$  is maximum. In other words, a majority vote is performed on the resulting classifications from the SVM classifier.

### 4. ATTACKS

In order to defeat the intrinsic printer identification system we design our attacks to mimic and obscure the underlying banding signals in the document. This is performed by preprocessing the document before it is sent to the printer with the addition of sinusoidal signals and gaussian noise.

The first attack adds a fixed amplitude fixed frequency sinusoidal signal

$$s_1(i) = \frac{A}{2} \left[ 1 + \cos \left( \frac{2\pi f i}{R_p} \right) \right] \quad (7)$$

to the printed regions of a document before sending it to the printer. Assume  $I(i, j)$  is an 8-bit grayscale document image to be printed. In the case of a text document,  $I$  will only take on values 0 and 255 corresponding to printed text or background respectively. The attacked document image can then be written as

$$\tilde{I}(i, j) = \begin{cases} I(i, j) + s_1(i) & , \quad I(i, j) = 0 \\ 255 & , \quad else \end{cases} \quad (8)$$

This attack is designed to mimic an intrinsic banding signal.

The second attack is a binarized version of the first. Since  $\tilde{I}$  above is grayscale, it will be automatically halftoned when it is printed by the printer driver or the printer itself. If for some reason it is not desirable to have the printer or printer driver perform the halftoning the attack function can be modified as follows. Let the sinusoidal signal define a threshold

$$s_2(i) = \frac{1}{16} \left[ 1 + \cos \left( \frac{2\pi f i}{R_p} \right) \right]. \quad (9)$$

For every  $I(i, j)$  equal to 0, generate a random number  $\gamma_{i,j}$  uniform in  $[0, 1]$ . Then define the attacked image as

$$\tilde{I}(i, j) = \begin{cases} 255 & , \quad I(i, j) = 0 \quad \text{and} \quad \{\gamma_{i,j} < s_2(i)\} \\ I(i, j) & , \quad else \end{cases} \quad (10)$$

The third attack uses a frequency hopping sinusoidal signal defined by

$$s_3(i) = \frac{A}{2} [1 + \cos(\phi(i))], \quad (11)$$

$$\phi(i) = \phi(i-1) + \frac{2\pi f(i)}{R_p}, \quad (12)$$

where  $f(i)$  is a randomly chosen frequency from from  $[30, 120]$ . Additionally,  $f(i)$  remains fixed over  $\alpha$  consecutive rows where  $\alpha$  is randomly chosen from  $[0, 100]$  with each frequency change. The attacked image then takes the same form as Equation 8 with  $s_3$  in place of  $s_1$ . This attack is designed to defeat the majority voting performed on the classifications of each individual “e”.

The fourth attack is a binarized version of the third using

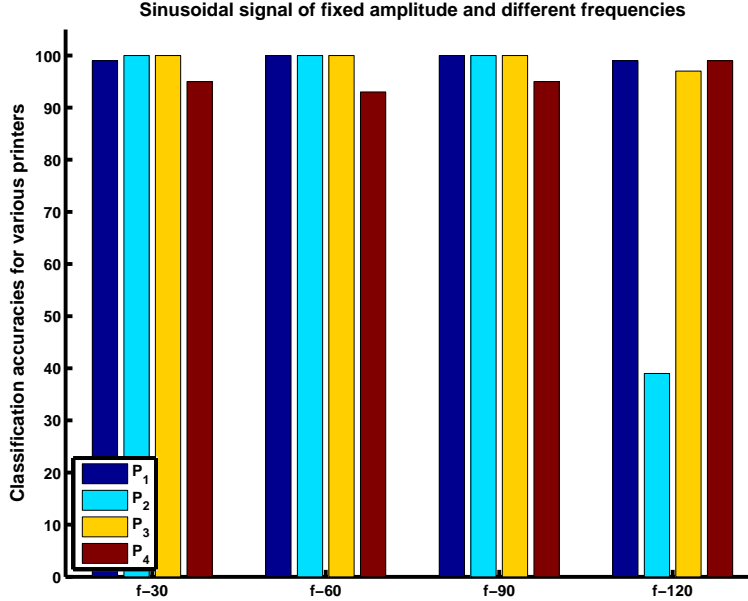
$$s_4(i) = \frac{1}{2^\nu} [1 + \cos(\phi(i))] \quad (13)$$

as the decision threshold on whether or not to turn a pixel white (as in Equation 10). The parameter  $\nu$  is an integer greater than zero that controls the threshold. Larger values of  $\nu$  will result in fewer pixel being flipped, or equivalently lower noise power.

The fifth attack adds Gaussian noise to the printed regions of the document image. In this case the attacked image is

$$\tilde{I}(i, j) = \begin{cases} I(i, j) + \mathbf{X} & , \quad I(i, j) = 0 \quad \text{and} \quad \mathbf{X} \in [0, 3\sigma] \\ I(i, j) & , \quad else \end{cases} \quad (14)$$

where  $\mathbf{X} \sim \mathcal{N}(\frac{3}{2}\sigma, \sigma^2)$ .



**Figure 5.** SVM classification results of individual “e”s after attack 1 (single frequency sinusoid).

## 5. EXPERIMENTS

A test page was created consisting of approximately 2000 “e”s in 12 point Times Roman font. A 17 page text document was generated consisting of 16 attacked versions of this test page. The first page contained no attack. Four pages contained attack 1 with parameters  $A = 50$  and  $f = \{30, 60, 90, 120\}$  respectively for the four pages. One page contained attack 2 with parameter  $f = 75$ . Four pages contained attack 3 with parameter  $A = \{25, 50, 75, 100\}$ . Three pages contained attack 4 with parameter  $\nu = \{2, 3, 4\}$ . The last four pages contained attack 5 with parameter  $\sigma = \frac{1}{3}\{25, 50, 75, 100\}$ .

The document was printed on each of the four printers listed in Table 1. Each was then scanned in 8-bit grayscale at 2400 DPI. The training set consisted of 1500 feature vectors from the non-attacked page only. The feature vectors from the remaining 16 (attacked) pages were used for testing using the SVM classifier.

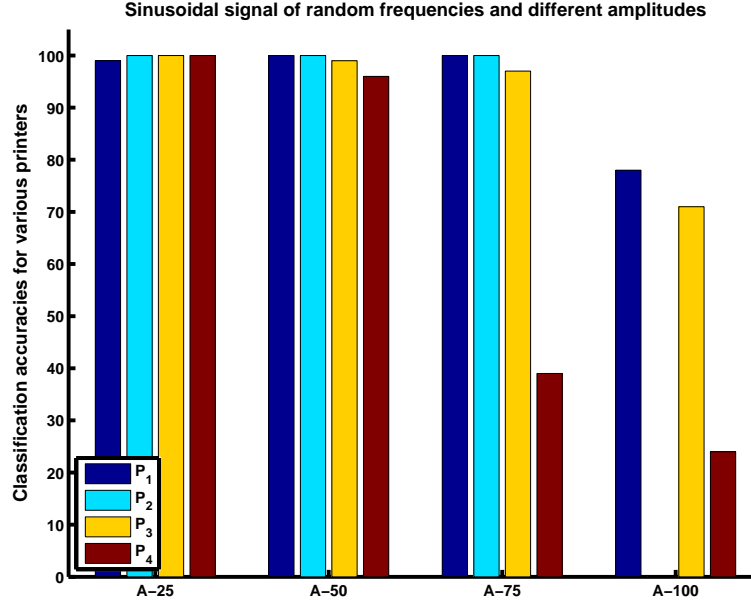
Since the SVM classifier is limited in mapping a feature vector only to one of the known training classes, it has limited scalability. If more printers are involved in training-testing then classification accuracies may drop significantly. Therefore, in addition to the SVM experiments, we perform linear discriminant analysis (LDA) and cluster analysis to determine which features are the most significant in each attack scenario. Since we know that the DFT features work best with non-saturated text (i.e. attacked text), and the GLCM works with saturated text (i.e. non-attacked text), we perform LDA on the GLCM and DFT features independently.

## 6. RESULTS

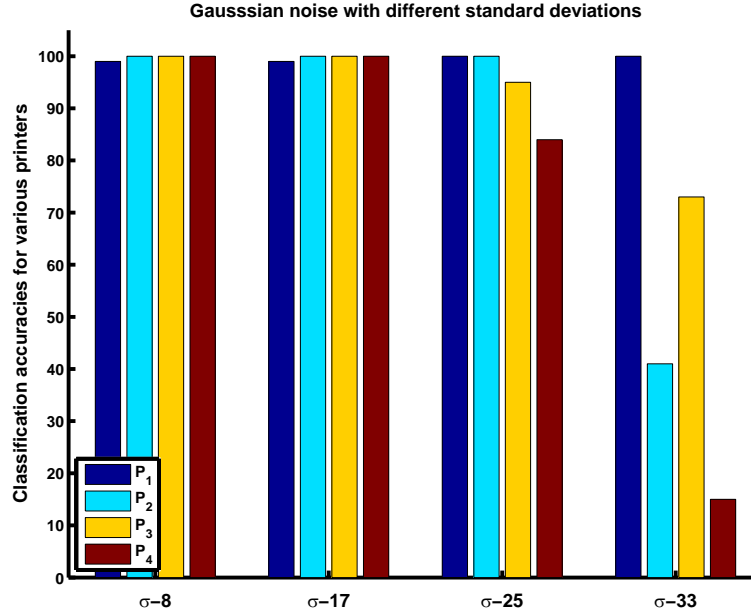
The graph in Figure 5 shows the classification accuracy of the individual “e”s in the test documents attacked with the single frequency sinusoidal signals of attack 1. We see that this attack does not have a large effect on the classification accuracy, except for  $P_2$  when  $f = 120$  cycles/inch.

The graph in Figure 6 shows the classification accuracy of the individual “e”s in the test documents attacked with the random frequency sinusoidal signals of attack 3. Here we see classification accuracy drop as the amplitude  $A$  goes above 75. However the classification accuracy still remains relatively high for  $P_1$  and  $P_3$ .

The results for attacks 2 and 4, the binarized version of attacks 1 and 3, show all printers at above 99% accuracy except for  $P_2$  which has more than 90% of its “e”s mis-classified as belonging to  $P_3$ .



**Figure 6.** SVM classification results of individual “e”s after attack 3 (random frequency sinusoid).



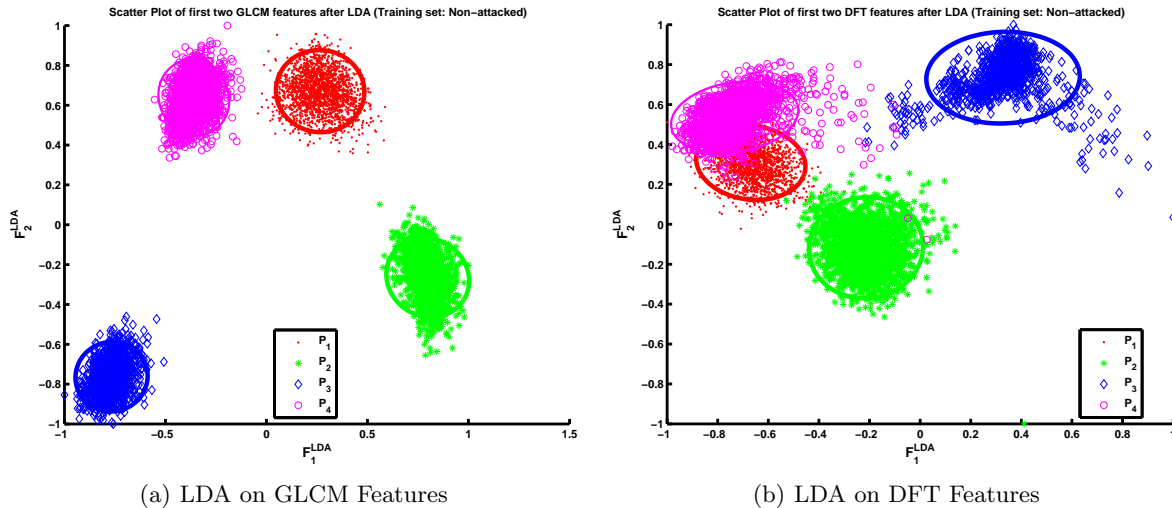
**Figure 7.** SVM classification results of individual “e”s after attack 5 (Gaussian noise).

The graph in Figure 7 shows the classification accuracy of the individual “e”s in the test documents attacked with the gaussian noise of attack 5. Classification accuracy starts to drop off after  $\sigma = 25$ , at which point the visual quality of the text also begins to decline.

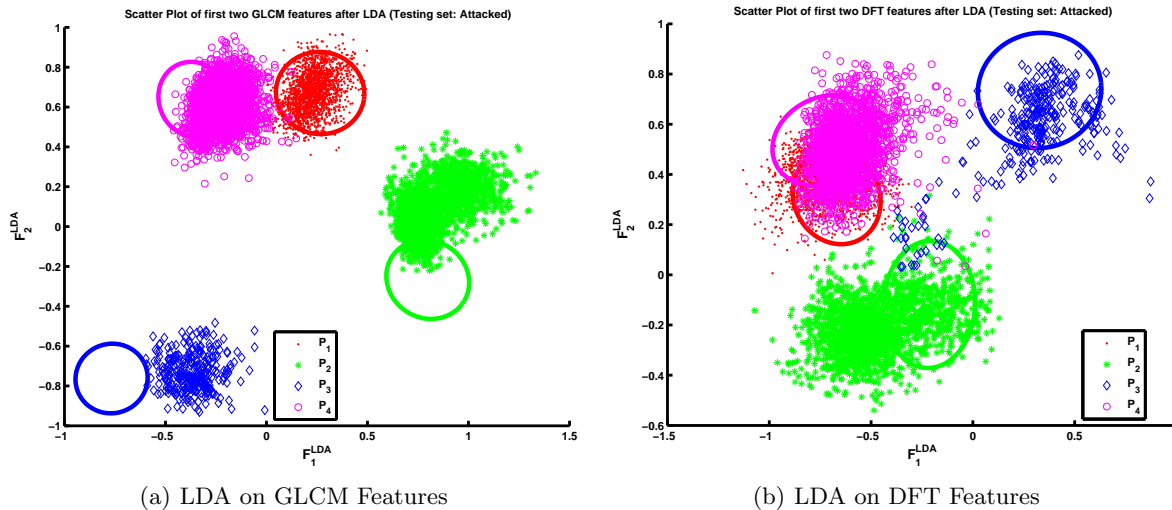
To gain a better understanding of how the features we have chosen behave in each of the attack scenarios, we look at the following plots in Figures 8 through 12. Each figure contains two subfigures each showing a scatter plot of the first two features from the reduced feature set obtained by applying LDA to the 24 GLCM and pixel features (subfigure ‘a’), and to the 15 DFT features (subfigure ‘b’). Figure 8 shows the non-attacked training



vectors. In each scatter plot, an oval is drawn for each class to represent the boundary within which 90% of that class' training vectors lie. These ovals are superimposed on Figures 9 through 12 to show how the features from that attacked page correspond to the original non-attacked features. We see from these figures that there is a large separation between clusters in at least one of the two reduced feature spaces, which suggests that in all cases we can identify the correct printer even when the printed page is perceptually different from the original. This shows that this method can be scaled up to more printers. Images of “e”s from for each of these attacks is shown in Table 2.



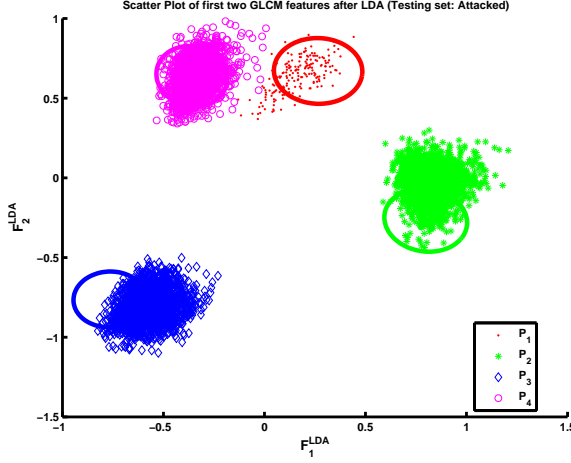
**Figure 8.** Scatter plot of first two LDA features from non-attacked “e”s. Ellipse identifies the boundary of the training cluster. All points are from the non-attacked document for that printer and are used to define the training cluster.



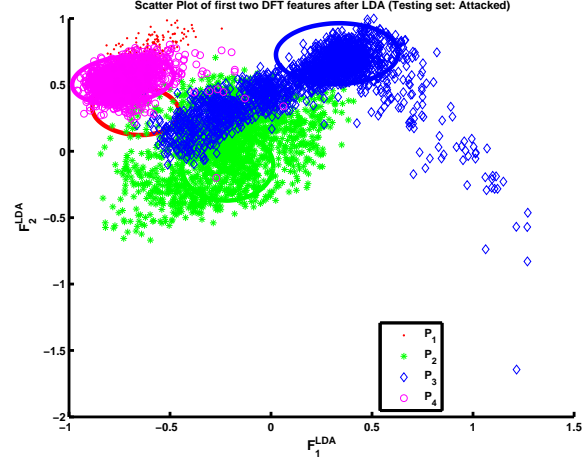
**Figure 9.** Scatter plot of first two LDA features for attack 1 (fixed sinusoid:  $A = 50$ ,  $f = 90$ ). Ellipse identifies the boundary of the training cluster. All other points are from the attacked document for that printer

## 7. CONCLUSION

The proposed modifications to intrinsic signatures for printer identification from text documents perform well under several attack models. The effectiveness of this system only starts to break down when the perceptual

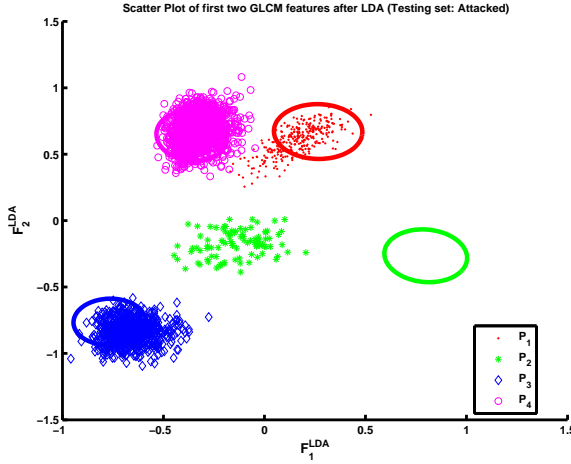


(a) LDA on GLCM Features

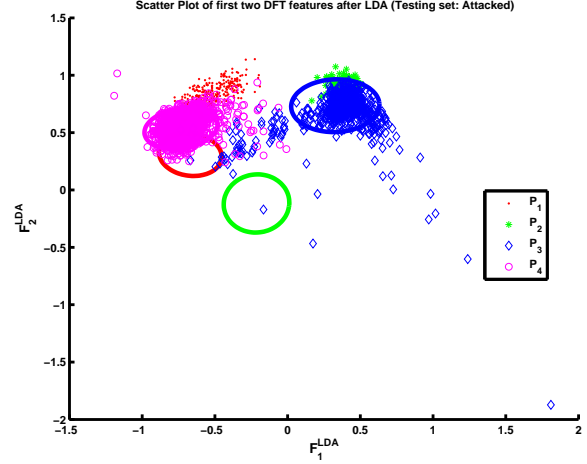


(b) LDA on DFT Features

**Figure 10.** Scatter plot of first two LDA features for attack 3 (random frequency sinusoid:  $A = 25$ ). Ellipse identifies the boundary of the training cluster. All other points are from the attacked document for that printer



(a) LDA on GLCM Features



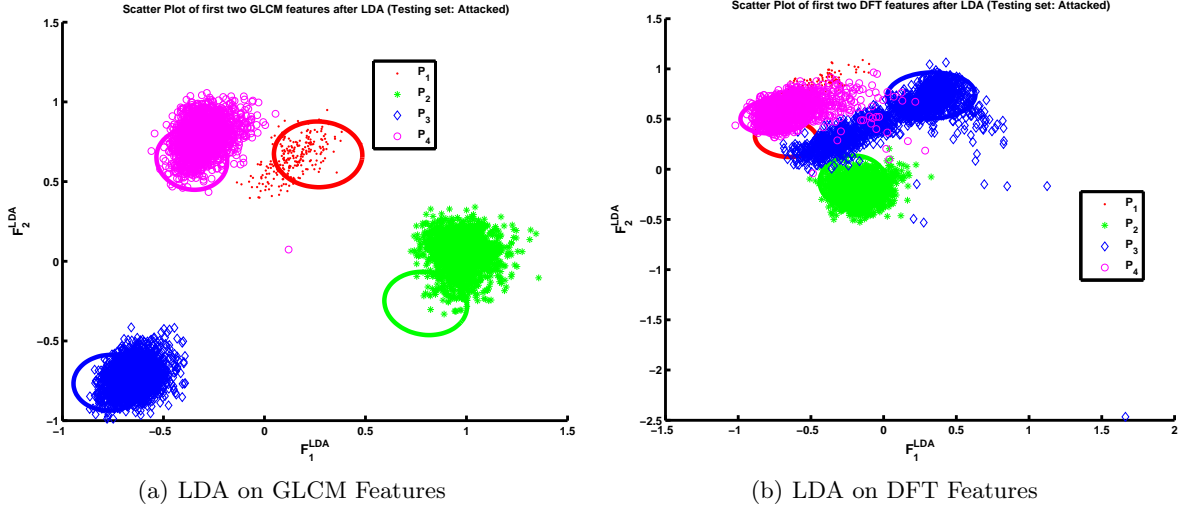
(b) LDA on DFT Features

**Figure 11.** Scatter plot of first two LDA features for attack 4 (random frequency binarized sinusoid:  $\nu = 4$ ). Ellipse identifies the boundary of the training cluster. All other points are from the attacked document for that printer

quality of the text is greatly reduced. As was shown in the results using the reduced feature sets, this method is scalable to a larger number of printers using a distance based classifier. Including more printing modes in the training set may also improve the results.

## REFERENCES


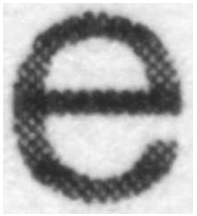


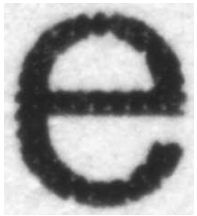
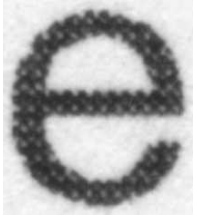











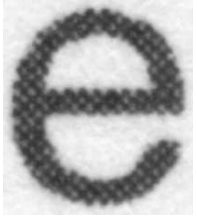
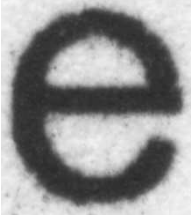









1. G. N. Ali, P.-J. Chiang, A. K. Mikkilineni, J. P. A. and George T.-C. Chiu, and E. J. Delp, "Intrinsic and extrinsic signatures for information hiding and secure printing with electrophotographic devices," *Proceedings of the IS&T's NIP19: International Conference on Digital Printing Technologies*, vol. 19, New Orleans, LA, September 2003, pp. 511–515.
2. A. K. Mikkilineni, O. Arslan, P.-J. Chiang, R. M. Kumontoy, J. P. Allebach, G. T.-C. Chiu, and E. J. Delp, "Printer forensics using svm techniques," *Proceedings of the IS&T's NIP21: International Conference on Digital Printing Technologies*, vol. 21, Baltimore, MD, October 2005, pp. 223–226.



**Figure 12.** Scatter plot of first two LDA features for attack 5 (Gaussian noise:  $\sigma = 8$ . Ellipse identifies the boundary of the training cluster. All other points are from the attacked document for that printer

3. G. N. Ali, P.-J. Chiang, A. K. Mikkilineni, G. T.-C. Chiu, E. J. Delp, and J. P. Allebach, "Application of principal components analysis and gaussian mixture models to printer identification," *Proceedings of the IS&T's NIP20: International Conference on Digital Printing Technologies*, vol. 20, Salt Lake City, UT, October/November 2004, pp. 301–305.
4. R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, November 1973.
5. R. W. Connors, M. M. Trivedi, and C. A. Harlow, "Segmentation of a high-resolution urban scene using texture operators," *Computer Vision, Graphics, and Image Processing*, vol. 25, pp. 273–310, 1984.
6. 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.
7. 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.
8. A. K. Mikkilineni, P.-J. Chiang, G. N. Ali, G. T. C. Chiu, J. P. Allebach, and E. J. Delp, "Printer identification based on graylevel co-occurrence features for security and forensic applications," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents VII*, vol. 5681, San Jose, CA, March 2005, pp. 430–440.
9. T. Joachims, "Making large-scale support vector machine learning practical," *Advances in Kernel Methods: Support Vector Machines*, B. Schölkopf, C. Burges, and A. Smola, Eds. MIT Press, Cambridge, MA, 1998.

Table 2.

Attack \ Printer	$P_1$	$P_2$	$P_3$	$P_4$
-				
1				
$A = 50, f = 90$				
3				
$A = 25$				
4				
$\nu = 4$				
5				
$\sigma = 8$				