Printer ballistics through texture analysis of characters

Adriano Ruggero[*]
Gabriel Rodrigues[†]
Mário Brito[‡]
Maurício Perez[§]
Anderson Rocha[¶]

Abstract

We describe a technique for ballistics of printed documents, that is, link a printed document to a specific printer. The principle of this technique is the analysis of character's texture, by extracting some properties from the characters of scanned images from the printed documents, and relate this properties through a co-occurrence matrix. This matrix can be used to create a "fingerprint" for the characters related to the same printer, what allows us to identify the specific printer device that printed these characters.

1. Introduction

In August of 2013, a russian man wrote his own small print in a credit card contract [1]. The credit card's administrator bank didn't read the amendments made by the client, and just signed and certified the document. The changes included unlimited credit line, 0 percent interest rates and no fees. When the bank decided to terminate the man's credit card, because overdue payments, he sued them for more than 24 million rubles (US$ 727.000). How could the bank prove the falsification?

Altough we are living in a digital era, printed documents still are a significant part of our day by day. Likewise, with the constant reduction in prices and increase in quality of printing equipment, forgeries become increasingly commonplace.

Legal aspects aside, a way to verify if a document, or a part of it, came from a specific device can be through character's texture analysis.

Our approach for the analysis of character's texture is as follows. From printed pages scanned at high resolution, selected characters were extracted. From these characters, we obtained its properties of contrast, correlation, energy and homogeneity, creating with them a co-occurrence matrix. This matrix can be called a "fingerprint" of the character. This "fingerprint" of characters is closely related to the printing device which originated it, and can be used to identify which printer was responsible for printing it.

However, slight imperfections may occur during the printing and/or scanning process of documents. To handle these small errors (or variations), characters were selected from different areas of scanned document, their properties were obtained and then classified using machine learning algorithms.

2. State-of-the-Art

Literature on the identification of printing devices from their printed documents is not wide in Academia. The work of Kee and Farid [2] reports a technique of geometric modeling of degradation caused by the printer. According to the authors, this degradation, measured based on a character basis, is modeled constructing a linear basis from characters commonly found on a printed page. From the point of view of ballistics, the accuracy of the developed technique allows to identify the device that originated the printed pages. According to the authors, the technique allows to distinguish between printers of different brands and models, but not among printers of the same make and model.

The work Bulan et al [3] demonstrates a technique for analyzing geometric distortions introduced during the printing process of electrophotographic printers (EP). According to the authors, it is first determined the geometric distortion signature to the printer using only printed pages. It is then built up a database and the printer used to print a test page is identified by computing the geometric distortion signature

---

[*]Institute of Computing, University of Campinas (Unicamp). **Contact**: `arruggero@lasca.ic.unicamp.br`

[†]Institute of Computing, University of Campinas (Unicamp). **Contact**: `gabriel_rodrigues@aol.com`

[‡]Institute of Computing, University of Campinas (Unicamp). **Contact**: `britomar@aedu.com`

[§]Institute of Computing, University of Campinas (Unicamp). **Contact**: `mauriciolp84@gmail.com`

[¶]Institute of Computing, University of Campinas (Unicamp). **Contact**: `anderson.rocha@ic.unicamp.br`

from the printed test page and correlated it with the signatures present in the database. Geometric distortion signature is calculated comparing dot positions extracted from the printed image and estimated dot positions before printing.

The work of Pollard et al [4] presents a general method for extracting a signature print from the outer boundary of a text glyph.

Valuable is also the work of Rocha and Leite [5]. This work deals with the concept of analysis of textures through co-occurrence matrix, used as a cornerstone of the solution proposed by us.

3. Proposed Solution

Our solution consist in getting the image of characters selected from scanned documents in grayscale, create a co-occurrence matrix, extract its properties of contrast, correlation, energy and homogeneity - creating a feature vector - and cluster them by machine learning algorithms.

3.1. Printers dataset

The documents used for this work were obtained from the Wikipedia site [6], and were written in English, some of them containing pictures, some not. They were printed by printers listed in Table 1.

These printed documents were scanned at high resolution and saved as Tiff file format (Tagged Image File Format), forming a database. These files were made available through an FTP site [7]. An example of a typical document used in this study can be seen in Figure 1.

3.2. Characters

The characters chosen for this work were "e" and "t", both in lowercase, because they are, respectively, the first and second most common letters in texts written in English [8].

To avoid inconsistencies and / or values "off the curve", we used only characters printed in the same font and size, for all printers. With the same objective, we used only characters printed in normal font, not analyzing characters in bold, italic, underline, strikeout, superscript, subscript, or any other form of letter change.

Were not considered characters from misaligned documents, i.e. characters from apparently obliquely scanned documents. Figure 2 shows examples of two characters obtained from scanned documents, one of them being usable and the other not, being considered rotated.

Subsection 3.4 will be explained in more detail why not use rotated characters.

Figure 1. Typical document used in this work [6].

Figure 2. Examples of characters: the left one was considered useful; the right one, slightly rotated.

3.3. Printers

Due to the fact that all scanned documents come from laser printers, it was necessary to take some precaution-

Table 1. Printers used in this work

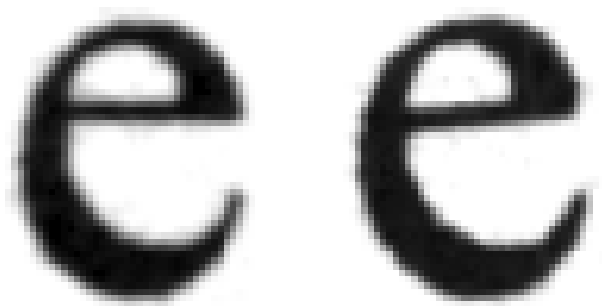| Printer	Documents	Characters "e"	Characters "t"
Brother-HL4070CDW	28	252	252
Canon-D1150	28	252	252
Canon-MF3240	28	252	252
Canon-MF4370DN	27	252	252
HP-CLJ-CP2025A	28	250	250
Lexmark-E260D	30	636	629

Figure 3. Default page printer schema [9].

Figure 4. Division of the scanned document in areas.
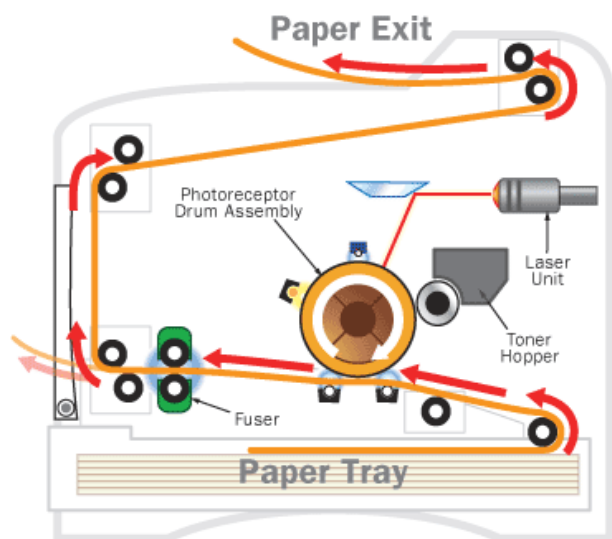
ary measures. Laser printers are known as "page printers", while dot matrix printers and inkjet printers are called "line printers". This is a crucial difference, and should be considered for more careful study.

Line printers print documents line by line from the top of the sheet, keeping a characteristic pattern which periodically repeats for each paper feed. That is, any line printed by this printer model will have basically the same characteristics, regardless of their vertical location in the paper sheet.

Page printers, instead, do not print documents line by line. In this case, an image of the entire page is "printed" on the photoreceptor drum by a laser unit. This image attracts toner particles, and then transfers it to the paper sheet. Finally, a fuser unit heats the paper, so the toner melts and attach it. Figure 3 shows a default page printer schema.

By working in this way, page printers do not maintain a characteristic pattern that is repeated line by line across the printed paper sheet. Due to small imperfections that may exist in the photoreceptor drum, each print area generated by this type of device can present different characteristics. Taking into account this fact, it was necessary to obtain characters from different parts of the scanned document.

Basically, each document has roughly divided into three parts: upper, middle, and bottom. The Figure 4 shows an example of a document divided in this way. If this article is being viewed in color, the upper part of the figure is in red, the middle, in green and bottom, in blue.

Despite showing subtle differences between characters located in different areas of the printout, the print device maintains certain intrinsic characteristics unchanged in all printed characters. Such characteristics can be compared to our fingerprints, making it a way to link a printed character to a particular device.

Table 2. Differences between an original character and a rotated character.

Property	Original character	Rotated character (-4º)
Contrast	3.2443 - 2.2905	5.1617 - 4.6504
Correlation	0.7869 - 0.8502	0.6967 - 0.7264
Energy	0.1608 - 0.1744	0.1106 - 0.1216
Homogeneity	0.6946 - 0.7462	0.6610 - 0.6995

3.4. Gray level co-occurence matrix

Co-occurrence matrix is a tabulation of how many different combinations of intensity values of pixels (grayscale) occur in an image. The primary use of the co-occurrence matrix is characterized texture in an image from a set of statistics for instances of each gray level in different pixels along different directions [5].

Co-occurrence matrix considers the relationship between two pixels at a time, one called as "reference pixel" and another as "neighbor pixel". The neighboring pixel chosen can be neighbor in any direction: east (right), west (left), north (above), south (below), or diagonally, i.e. northeast, northwest, southwest and southeast of each reference pixel. Also the neighborhood need not be exactly one pixel, can be 2, 3, or any value. Each pixel within the image becomes the reference pixel, starting at the top left and proceeding to the lower right. There will be particular cases. For instance, pixels of the right margin of the image does not have neighboring right.

Co-occurrence, in its most general form, can be specified by a matrix of relative frequencies $P(i, j, d, \theta)$, where two neighboring texture elements separated by a distance d in a direction θ occur in the image, one with the property i and another with property j.

From a co-occurrence matrix can be obtained some properties of the analyzed image. For this work, we used the properties of contrast, correlation, energy and homogeneity to create a fingerprint of the printed characters.

Using an example similar to the rotated character, we can compare the value differences in the properties obtained from the co-occurrence matrix (Figure 5). In this case, the character on the left was obtained directly from the scanned document; character on the right is the same character, but rotated -4º.

Table 2 lists the differences between the properties of the character sample obtained from the co-occurrence matrix.

As shown in the Table 2, small changes in the alignment of the print / scan can cause large variations in the properties values obtained from the co-occurrence matrix.

Details on how to calculate values of contrast, correlation, energy and homogeneity can be seen in Appendix A.
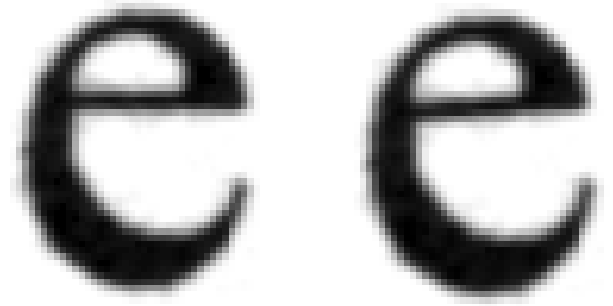
Figure 5. On the left, a character obtained from a scanned document and to the right, the same character, rotated -4º.

4. Experiments and Discussion

As a starting point for our experiments, it was necessary to select the desired characters from the documents on which the properties would be extracted. With the extracted properties of contrast, correlation, energy and homogeneity from the characters, we started evaluating the best methodology and machine learning algorithms for our solution. In the first instance, we needed to determine the best options for classifying the characters and attributing them to a Printer. But our final goal is not to attribute the characters to a printer, but a whole document, so, in the next step, we propose a way of attributing the printer based on the classification of the characters.

4.1. Characters Selection and Properties Extraction

First we selected three characters from each one of the three sections from a document, and repeated that for all documents. There were 3x3x28 = 252 characters selected for each printer, what gave us a total of 1512.

For the properties extraction we used two different neighborhoods show in figure 6. From the first neighborhood we gathered two characteristics per type (contrast, correlation, energy and homogeneity), originating a feature vector of eight values for a character. The second one gave us four per type, in other words, a sixteen sized vector.

4.2. Characters Classification

The just acquired features are used for evaluation the available classification algorithms. For this evaluation it was separated 66% of the chars to be used as training (1008) and the rest for testing (504). From them we separate the most prominent and present the results in table 3.

Figure 6. Neighborhood A, leftmost, and B, rightmost, used in properties extraction (the darker pixel is the pixel-of-interest).

Table 3. Percentage of correct classifications of printers.

Method	Neighborhood A		Neighborhood B	
	Chars e's	Chars t's	Chars e's	Chars t's
Logistic	81	81.3	85	84.6
KStar	77.6	83	72	79.6
RotationForest	83.1	85	81.7	85.7
NNge	74.1	80.2	72.2	67.8
LMT	83.8	84.6	82.7	85.5

Table 4. Percentage of correct printer attribution.

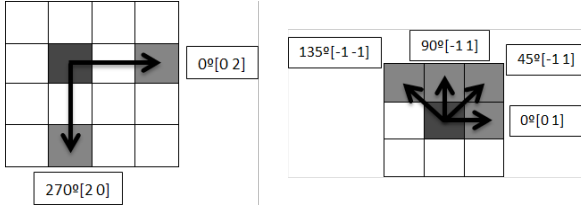| Method	Chars e's	Chars t's	Chars e's and t's
Logistic	21/24 = 87.5%	22/24 = 91.7%	21/24 = 87.5%
RotationForest	21/ 24 = 87.5%	21/24 = 87.5%	22/24 = 91.7%
LMT	21/24 = 87.5%	21/24 = 87.5%	22/24 = 91.7%

Analyzing the results we can see a slightly better classification on the characters "t" (perhaps they are more unique for each printer). The results also show that using a neighborhood of four pixels, for many cases, made the classification worst, what was against our intuition, since its descriptors were longer than of the other neighborhood we used.

4.3. Printer Attribution

As said before, attributing a printer to the character is not the main objective, so it was necessary to define a logical output from the classified characters obtained in the previous step. For this purpose we choose to attribute a document to the printer that had most classified characters on the document sample of characters.

Since there were no significant difference between Neighborhoods A and B, we tested using only the first. For this tests it were separated 4 documents per printer, 24 in total, and the rest of the documents were used for training. We choose to use only the three classifier methods that had the best results in characters classification. The printer attribution results are present in table 4.

Again, it is shown that using only characters "e" or only characters "t" we have almost the same correct classification rate. If we combine the two characters classification and use it for attribution there is a small improvement on the results.

5. Conclusions and Future Work

In this article we have described a technique for printer attribution based on the properties extracted from the character's texture of a document. These properties were contrast, correlation, energy and homogeneity, and could be used to create a signature for the printer that generated the document.

The feature vector size, formed by these properties, varies depending on the neighborhood defined for the generation of the Co-occurrence matrix. In the experiments made we used two different neighborhoods, and it was shown that a greater feature vector doesn't mean we will have a better classifier. Still, there is room for more tests in this aspect, experiments with other neighborhoods size and in different directions must be done for identifying the best parameters. Another consideration that can be taken in account is if all properties extracted from the co-occurrence matrix should be used, perhaps different combinations of them can bring better results.

For the experiments done, we used only the characters "e" and "t". Anyhow, the results made us believe that the character choice would have probably low impact on the results obtained, since in some methods for classification there is a higher accuracy and for others a lower when we alternate between "e" and "t". What can have a better impact is the number of characters selection from each part of the document (we picked only a small portion of them).

Another direction can also be made using other texture analyzers, such as HOG (Histograms of Oriented Gradients Extract) or LBP (Local Binary Patterns), and no GLCM.

Our work presents an unexplored characteristic from the printers uniqueness printing process that can be extracted afterwards with the scanned documents created on them. It was also shown that this characteristic can be used in ballistics for printer attribution but that still needs further investigation and experiments on the best methodology for achieving optimal results.

Although not the purpose of this study, and future studies, manual extraction of characters is still a counterproductive way of working. Develop a (semi) automated method would be of great benefit to researchers in this field.

6. Acknowledgements

References

[1] RT. 700k windfall: Russian man outwits bank with handwritten credit contract. http://rt.com/business/

`man-outsmarts-banks-wins-court-221/`, last access in November 17, 2013. 1

[2] Eric Kee and Hany Farid. Printer profiling for forensics and ballistics. In *Proceedings of the 10th ACM Workshop on Multimedia and Security*, MM&Sec '08, pages 3–10, New York, NY, USA, 2008. ACM. 1

[3] Orhan Bulan, Junwen Mao, and Gaurav Sharma. Geometric distortion signatures for printer identification. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2009, 19-24 April 2009, Taipei, Taiwan*, pages 1401–1404. IEEE, 2009. 1

[4] Stephen B Pollard, Steven J Simske, and Guy B Adams. Model based print signature profile extraction for forensic analysis of individual text glyphs. In *Information Forensics and Security (WIFS), 2010 IEEE International Workshop on*, pages 1–6. IEEE, 2010. 2

[5] Anderson de Rezende Rocha and Neucimar Jerônimo Leite. Classificação de texturas a partir de vetores de atributos e função de distribuição de probabilidades. 2, 4

[6] Wikipedia. Wikipedia - the free encyclopedia. `http://www.wikipedia.org/`, last access in November 19, 2013. 2

[7] Giuliano Pinheiro. Index of / giulianorp/printer_dataset. `http://www.recod.ic.unicamp.br/~giulianorp/printer_dataset/`, last access in November 19, 2013. 2

[8] University of Notre Dame. Letter frequencies in the english language. `http://www3.nd.edu/~busiforc/handouts/cryptography/Letter.html`, last access in November 17, 2013. 2

[9] Forensic Document Examination Services. Letter frequencies in the english language. `http://www.forensicdocumentexaminer.co.uk/Document_Analysis.html`, last access in November 17, 2013. 3

A. Formulas

p represents the pixel-of-interest.
i and j represents the value of the properties.
μi and μj represents the mean of i and j, respectively.
σi and σj represents the standard deviation of i and j, respectively.

A.1. Contrast

Returns the measure of contrast between the pixel of interest and neighboring pixel. For a constant image (same shade of gray to the full extent) the value is 0.

$$\sum_{i,j} |i-j|^2 p(i,j)$$

A.2. Co-relation

Returns the measure of how much the pixel is related to its neighbor. The range is from -1 to 1, where 1 means a fully correlated and -1, totally uncorrelated image.

$$\sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma_i \sigma_j}$$

A.3. Energy

Returns the sum of squared raised elements within the array. The range of possible values is from 0 to 1. A constant image (same shade of gray in all its extension) has value 1.

$$\sum_{i,j} p(i,j)^2$$

A.4. Homogeneity

Returns a value that represents the relative proximity of the diagonal elements of the co-occurrence matrix. The range of possible values is between 0 and 1, and the value 1 represents a diagonal matrix.

$$\sum_{i,j} \frac{p(i,j)}{1+|i-j|}$$

B. Features vectors

B.1. Training set

Figure 7 shows a portion of the file with the feature vectors for training the classifiers.

Figure 8 shows a portion of the file with the feature vectors for validation.

```
@relation glcm

@attribute Contrast_2_0 numeric
@attribute Contrast_0_2 numeric
@attribute Correlation_2_0 numeric
@attribute Correlation_0_2 numeric
@attribute Energy_2_0 numeric
@attribute Energy_0_2 numeric
@attribute Homogeneity_2_0 numeric
@attribute Homogeneity_0_2 numeric
@attribute Printer { Canon-MF3240,Canon-MF4370DN,Lexmark-E260D,Brother_HL4070CDW,Canon_D1150,HP-CLJ-CP2025A }

@data
0.935450,1.730948,0.909656,0.834475,0.476793,0.449284,0.894739,0.840638,HP-CLJ-CP2025A
0.944974,1.664401,0.904148,0.832784,0.489461,0.452085,0.895935,0.845581,HP-CLJ-CP2025A
0.905115,1.512701,0.906148,0.844743,0.509096,0.479459,0.899327,0.848420,HP-CLJ-CP2025A
1.108289,1.790340,0.888423,0.821509,0.454398,0.417157,0.875183,0.821525,HP-CLJ-CP2025A
0.962257,1.661538,0.906548,0.840232,0.481362,0.451634,0.888355,0.830250,HP-CLJ-CP2025A
1.077249,1.804651,0.895807,0.827121,0.448186,0.416945,0.876749,0.821001,HP-CLJ-CP2025A
0.891711,1.620751,0.912642,0.842786,0.484403,0.454163,0.892011,0.843035,HP-CLJ-CP2025A
0.911464,1.526297,0.908290,0.847974,0.493627,0.463476,0.893333,0.836255,HP-CLJ-CP2025A
0.898765,1.791771,0.913720,0.829508,0.470888,0.429804,0.893633,0.834404,HP-CLJ-CP2025A
0.905820,1.716637,0.912562,0.836138,0.474820,0.427315,0.902095,0.825540,HP-CLJ-CP2025A
0.868430,2.275492,0.919578,0.791215,0.450482,0.392750,0.887025,0.802319,HP-CLJ-CP2025A
0.897354,1.545617,0.911566,0.849104,0.481332,0.446972,0.901975,0.839815,HP-CLJ-CP2025A
0.989065,1.757424,0.900178,0.824361,0.488990,0.455758,0.897437,0.853104,HP-CLJ-CP2025A
0.860670,1.693023,0.911620,0.827874,0.497883,0.466820,0.899365,0.851920,HP-CLJ-CP2025A
```

Figure 7. Feature vector for training.

```
@relation glcm

@attribute Contrast_2_0 numeric
@attribute Contrast_0_2 numeric
@attribute Correlation_2_0 numeric
@attribute Correlation_0_2 numeric
@attribute Energy_2_0 numeric
@attribute Energy_0_2 numeric
@attribute Homogeneity_2_0 numeric
@attribute Homogeneity_0_2 numeric
@attribute Printer { Canon-MF3240,Canon-MF4370DN,Lexmark-E260D,Brother_HL4070CDW,Canon_D1150,HP-CLJ-CP2025A }

@data
0.895238,1.628623,0.917227,0.850874,0.473382,0.437550,0.891089,0.823679,HP-CLJ-CP2025A
0.966490,1.712343,0.905927,0.834918,0.463795,0.423079,0.895387,0.830467,HP-CLJ-CP2025A
0.840212,1.476923,0.915371,0.852687,0.474248,0.447473,0.889095,0.844448,HP-CLJ-CP2025A
0.902293,1.701610,0.916623,0.844267,0.465295,0.423845,0.888503,0.816416,HP-CLJ-CP2025A
0.881834,1.579964,0.913989,0.847263,0.422460,0.371630,0.876520,0.807436,HP-CLJ-CP2025A
0.881129,1.703041,0.917234,0.841612,0.498711,0.463248,0.899130,0.837329,HP-CLJ-CP2025A
0.838095,1.516279,0.915481,0.848647,0.503180,0.474335,0.897925,0.838795,HP-CLJ-CP2025A
0.969312,1.804651,0.908311,0.831012,0.505294,0.471849,0.895588,0.844620,HP-CLJ-CP2025A
0.879012,1.549911,0.911920,0.846234,0.497728,0.467111,0.902340,0.845224,HP-CLJ-CP2025A
0.961552,1.672272,0.908334,0.842157,0.500162,0.470821,0.896914,0.849189,Lexmark-E260D
0.878307,1.598569,0.910086,0.838007,0.519994,0.493258,0.912440,0.871360,Lexmark-E260D
0.994004,1.622182,0.905031,0.846562,0.504759,0.473760,0.894722,0.844514,Lexmark-E260D
0.967196,1.750268,0.908793,0.836591,0.501976,0.473303,0.895655,0.848221,Lexmark-E260D
0.870547,1.629338,0.915658,0.843703,0.500357,0.468077,0.896379,0.852564,Lexmark-E260D
0.928395,1.534168,0.907141,0.848070,0.495640,0.467214,0.903880,0.850948,Lexmark-E260D
1.106878,1.825403,0.901327,0.838870,0.494592,0.469842,0.893767,0.848429,Lexmark-E260D
0.915697,1.604293,0.912779,0.848687,0.483383,0.455584,0.893721,0.838014,Lexmark-E260D
1.001764,1.617174,0.903676,0.846044,0.498763,0.473696,0.898193,0.853268,Lexmark-E260D
1.904321 , 3.387175 , 0.869342 , 0.770757 , 0.205069 , 0.155162 , 0.805745 , 0.668047 , Brother_HL4070CDW
1.800727 , 3.144165 , 0.878520 , 0.790854 , 0.215290 , 0.171693 , 0.802109 , 0.673938 , Brother_HL4070CDW
```

Figure 8. Feature vector for validation.