Robustness of character recognition techniques to double print-and-scan process

Iuliia Tkachenko, Petra Gomez-Krämer

Robustness of character recognition techniques to double print-and-scan process

Iuliia Tkachenko and Petra Gomez-Krämer
L3i Laboratory, University of La Rochelle, Avenue Michel Crépeau, 17042 La Rochelle, France
{iuliia.tkachenko, petra.gomez}@univ-lr.fr

***Abstract*— The integrity check of printed and scanned documents is a hot topic these days. Several solutions were proposed for documents printed and scanned once. However forged documents quite often pass through a double Print-and-Scan (P&S) process. The P&S process impacts a lot the shape and color of characters. Therefore, the top OCR Engines cannot correctly recognize these characters. In this paper, we present the problems that the Tesseract OCR Engine faces with when trying to recognize the characters printed and scanned twice. We suggest to use the PCA based character recognition method that outperforms the Tesseract OCR in our experiments. We also show that the use of a pre-procesing step can improve the recognition results of double printed and scanned documents. Finally, we discuss the pros and cons of the PCA based recognition method.**

***Keywords*— double print-and-scan process, character recognition, printed document authentication, OCR robustness**

I. Introduction

Due to accessibility and cheapness of production devices such as printers, scanners and photocopiers the number of counterfeited documents increases. The administrations and companies face with a huge amount of forged electricity and phone bills, birth certificates and administrative invoices [11]. These documents can be presented either in hardcopy or in digital format. Therefore, the protection techniques differ depending on document format.

The digital documents are protected by signature algorithms (for example, the Digital Signature Algorithm developed by the National Institute of Standards and Technology) that are very efficient tools to ensure the electronic document authenticity. The hardcopy documents are often protected by watermarks [9], [6], fingerprints or copy-sensitive graphical codes [1]. Nevertheless, all these protection techniques authenticate the document support and cannot ensure the document integrity check.

In daily life we use the both formats of the same document. This type of documents that can be represented by hardcopy or digital format is called hybrid documents. One of the first hybrid protection systems was presented in [15]. The authors proposed two techniques to hash the text in electronic and printed documents. The first one uses the Optical Character Recognition (OCR) software and a classical cryptographic message authentication code. The second one is based on a random tilling text hashing technique that computes the mean luminance values of several random rectangles for each character or word. The first technique performs better than the second one, but the number of experiments was limited. The authors in [13], [14] suggest to extract the specific feature code from each character and construct the document hash using these features. The experiments show that the proposed scheme can resist to affine transformations, JPEG compression and low-level noise, but is not robust to median filtering. Additionally, the pre-processing step removes the punctuation from the text.

One of the recent studies of OCR capabilities [3] shows that the Tesseract algorithm (which is an open-source OCR) precision depends on the printer and scanner resolutions and varies from 91.58% to 99.45%. This precision brings the collision probability (when two different characters are recognized as the same) up to 0.002. Therefore, the use of a cryptographic hash function directly after OCR algorithm cannot be done for printed document signature construction.

When we talk about hybrid documents we can imagine two types of counterfeiting processes. During the first one, the digital document is changed and then either used in digital or hardcopy format. The second type consists of a hardcopy document change, for this an opponent needs to scan the document, to change the content and to reprint it. This type of counterfeiting implies a double Print-and-Scan (P&S) process applied to document. The character recognition of double printed documents is more difficult due to the double impact of the P&S process. Thus, the document hash construction is also a difficult task. In this paper we aim at studying the robustness of the Tesseract OCR Engine and the character recognition technique based on Principal Component Analysis (PCA) to the double P&S process. The PCA approach was used for handwriting character recognition, but it was not used in OCR techniques before.

The rest of the paper is organized as follows. We describe the P&S process and its impact to document image quality in Section II. The short overview of Tesseract OCR and the problems faced after double P&S process are presented in Section III. The character recognition system based on PCA as well as the pre-processing operation are introduced in Section IV. We compare the results in Section V. Finally, we conclude and discuss the future paths in Section VI.

II. Print and scan impact

Two hardcopy exemplars of the same document differ from each other in digital sense, but they are considered as the same by naked eyes. The comparison of these documents can

be done when they are digitized using a scanner. Therefore, the printing and scanning processes are not separable from each other and the distortions always belong to both of them [16]. In this section we aim at discussing the main degradations introduced by P&S process.

The modifications added by the P&S process can be produced by ink dispersion in the paper, inhomogeneous lighting conditions during the scan acquisition, resampling inherent to the P&S process or varying speed of the scanning device during the acquisition [1]. The impact of the P&S process is often considered as a physical unclonable function [5], due to the physical changes and the stochastic nature of these changes.

Each pair of printer-scanner produces the unique signature that can be used for printer or scanner identification (printer/scanner forensics). These forensic methods use the character contour degradation and noise added during the P&S process [10], the graylevel co-occurrence textured features and pixel based features [7], and the features based on convolutional texture gradient filters [4]. These research works and the P&S process modeling show that the printed and scanned documents need to be pre-processed before passing to the character recognition stage.

As printers use black ink or toner, the visual sensation of a gray level is obtained by creating a binary textured image. This operation, called halftoning, is specific to each printer brand. The resolution of the printer is also a factor of image degradation. It is measured in dots per inch (dpi). We illustrate the changes of the character 'a' after different P&S precesses in Fig. 1. We note that the black color after P&S process becomes gray. The contour after P&S at 300 dpi is much less sharp in comparison with a character printed at 600 dpi (see Fig. 1.b and Fig. 1.c respectively). And finally, the character that passes the double P&S process (Fig. 1.d) has lost the contour sharpness and the color homogeneity due to printer halftoning and automatic scanner processing.

(a) (b) (c) (d)

Fig. 1. The comparison of the character 'a': a) numeric, b) printed and scanned in 300 dpi, c) printed and scanned in 600 dpi, d) double printed and scanned in 600 dpi.

Quite often the OCR techniques take into account the reproduction problems (cropped, broken and filled characters), but they do not take into account the additional noise added to the document image as well as the distortions added after several P&S processes.

III. Existing techniques for character recognition

There exist a lot of open-source and commercial OCR systems. The best known and efficient open-source OCR is Tesseract [1]. In this section, we make a short overview of Tesseract system and discuss the problems we face with when using Tesseract for double P&S document recognition.

A. Overview of the Tesseract system

Tesseract is an OCR Engine originally developed by HP from 1985 till 1995, and from 2006 mostly developed by Google. The development history and overall architecture of Tesseract are presented in [12].

The pre-processing steps of Tesseract are adaptive thresholding and page layout analysis. The binarization threshold is calculated by using Otsu's method [8]. The Tesseract binarization process is not the most effective one, therefore, the several recommendations to improve the quality of the output can be found on Tesseract wiki-page [2]. After the pre-processing steps the system goes to word recognition, where the words are segmented in order to extract the character features and to feed the character classifier. The Tesseract uses the minimalist approach to find the best match for a character combination and looks for the minimal distance between the word and the dictionary.

During the word segmentation the bounding boxes of characters are extracted. These boxes are used for character feature extraction and match. Therefore, the extracted bounding boxes influence the character recognition accuracy.

B. Bounding box problems

In this section we want to discuss several problems that Tesseract faces with when processing the double printed and scanned document. The examples of bounding boxes extracted by Tesseract are illustrated in Fig. 2. Fig. 2.a illustrates an ideal example of bounding box that helps to correctly recognize the character. Fig. 2.b shows an example of non-centered bounding box. Due to the noise added by the P&S process small points can appear in the background. Tesseract considers these points as a part of character structure. Nevertheless, even if the bounding box is incorrect, the character is quite often correctly recognized in such bounding boxes. Fig. 2.c-d show the examples of bounding boxes with cropped characters. The number of such bounding boxes is quite big, additionally, Tesseract cannot recognize these characters correctly. For example, the bounding boxes shown in Fig. 2.c-d were recognized as character 't'.

(a) (b) (c) (d)

Fig. 2. Examples of extraction of bounding boxes of the character 'n' after a double P&S process: a) good bounding box, b) bounding box with big number of background lines in the top due to the noise point, c) cropped bounding box with leak of small amount of pixels, d) cropped bounding box with leak of part of character.

Due to the huge impact of P&S process to the character structure, the number of incorrect character recognition is

[1] https://github.com/tesseract-ocr

[2] https://github.com/tesseract-ocr/tesseract/wiki/ImproveQuality

very big in comparison with documents printed and scanned once. For example, we visualize the bounding boxes of characters that were recognized as letter 'e' in Fig. 3. This example shows that the errors can be unexpected as the difference of structure between letters 'e' and 'n' is huge. This type of errors is possible due to the use of dictionary and the search of minimal distance between the words. We note that these errors are not produced when we use the digital document or the document after one P&S process.

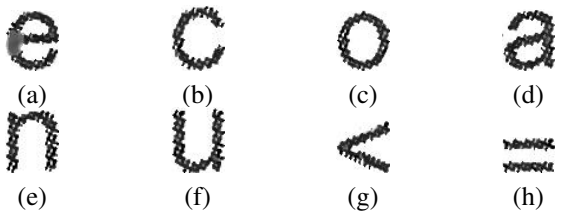

Fig. 3. The bounding boxes detected as letter 'e' after double P&S process by the Tesseract 3.05 OCR algorithm.

Thus, the errors of bounding box extraction produce a lot of unexpected recognition errors for double P&S documents. Our analysis of this problem is presented in Section V-B.

IV. Recognition system based on PCA

The recognition system based on the use of PCA classification [2] is well known. It was used several times for handwritten character recognition, but it has never been used for OCR. However, the handwriting characters and the characters after double P&S process, both have unstable shapes and unpredictable noise around the character, thus we suppose that the PCA can improve the recognition of characters after double P&S process. Additionally, this technique does not use the dictionary and can reject the unknown images. We present this recognition system and propose a pre-processing operation in this section.

A. Overview of PCA classification

The PCA based classification is a classical example of supervised learning techniques. It has a training and a classification step.

The training database is represented by square bounding boxes extracted from digital image. Let $I_i, i = 1, \cdots, M$ be the training database, where each character is represented by only one image I_i of size $N \times N$ pixels. The training stage consists of the following steps:

1) Pass the image matrix to vector form $\Gamma_i, i = 1, \cdots, M$ of size $1 \times N^2$.
2) Construct the matrix $\Gamma = [\Gamma_1, \cdots, \Gamma_M]$.
3) Normalize the image-vectors Γ_i by subtraction of the average image from the image-vectors, i.e. $\Phi_i = \Gamma_i - \Psi$.
4) Construct the matrix $A = [\Phi_1, \cdots, \Phi_M]$.
5) Calculate the eigenvectors $u_i, i = 1, \cdots, M$ of the covariance matrix $C = A \cdot A^T$.
6) Represent each training base image as a linear combination of all eigenvectors:

$$I_i = \Psi + \omega_1 \cdot U_1 + \cdots + \omega_M \cdot U_M,$$

where $\Omega_i = [\omega_1, \cdots, \omega_M]$ is a weight vector and $U_i, i = 1, \cdots, M$ is the eigenvector i in matrix representation (of matrix size $N \times N$).

During the classification of the input image I' of size $N \times N$ pixels the following steps need to be done:

1) Convert I' to the image-vector Γ' of size $1 \times N^2$.
2) Normalize this image-vector $\Phi' = \Gamma' - \Psi$.
3) Project the normalized image-vector onto the eigenspace and find the weight vector Ω'.
4) Calculate the distance between the input weight vector Ω' and all weight vectors of training set $\Omega_i, i = 1, \cdots, M$.

The minimal distance indicates the class that the image I' belongs to.

This classification method has an interesting option. We can determine the threshold in order to reject unknown images for our training database. This option can be very useful for character recognition as the extraction of bounding boxes is sometimes inaccurate as shown in Section III-B.

B. Pre-processing operation

As it was mentioned in Section II the double P&S process impacts more to character shape, therefore, the pre-processing is needed to improve the character recognition results. In order to fill the color inhomogeneity of character after double P&S process we suggest to use morphological operations: the 2×2 open-close operation. This pre-processing step improves the quality of characters and helps to better recognize the characters that suffer from double P&S process. A comparison of character after double P&S operation, after Otsu's binarization and after proposed pre-processing operation is illustrated in Fig. 4.

Fig. 4. The comparison of the character 'h': a) after double P&S process, b) binarized using Otsu's method, c) after proposed morphological pre-processing.

V. Experimental results

In this section we describe the database used, the errors produced by Tesseract during bounding box extraction and compare the Tesseract and the PCA based classification results for character recognition. Finally, we discuss the results while using the rejection threshold for PCA based classification and present pros and cons of this approach.

A. Database description

In our experiments we use 22 Arial font documents with a font size of 12. These documents were printed and scanned with a Konica Minolta Bizhub 223 printer-scanner at 300 dpi and 600 dpi resolutions. The double P&S documents were created by two successive P&S processes using the same printer-scanner device at 600 dpi resolution.

In our experiments, we used the bounding boxes only for lowercase characters. The number of bounding boxes varies

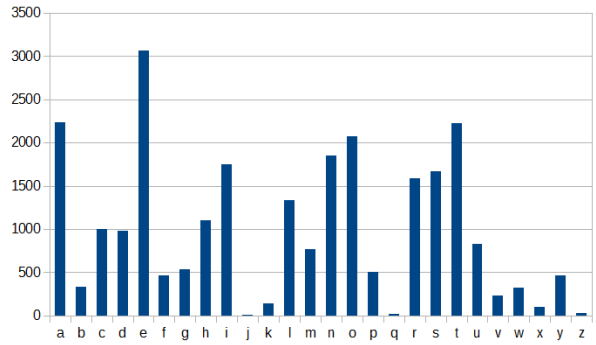

Fig. 5. The total number of bounding boxes for each character.

depending on the character use frequency. The total number of bounding boxes per character is illustrated in Fig. 5. We note that the smallest number of samples are found for letter 'j' - 13 items, letter 'q' - 16 items and letter 'z' - 28 items. All other characters have a statistically significant number of samples (between 142 and 3060 items). All these bounding boxes are well centered and do not have cropped or big background parts.

	Correct	Incorrect	Anomaly
letter a	98.68%	1.13%	0.18%
letter b	99.70%	0.30%	0.00%
letter c	97, 02%	2.98%	0.00%
letter d	97.70%	0.10%	2.20%
letter e	95.64%	4.05%	0.31%
letter f	96.20%	2.46%	1.34%
letter g	99.08%	0.37%	0.55%
letter h	99.46%	0.27%	0.27%
letter i	82.15%	16.00%	1.85%
letter j	100%	0.00%	0.00%
letter k	100%	0.00%	0.00%
letter l	96.56%	1.87%	1.57%
letter m	94.95%	0.00%	5.05%
letter n	97.66%	0.48%	1.86%
letter o	98.74%	1.26%	0.00%
letter p	100%	0.00%	0.00%
letter q	100%	0.00%	0.00%
letter r	99.06%	0.06%	0.88%
letter s	98.64%	1.10%	0.26%
letter t	96.30%	2.39%	1.31%
letter u	99.73%	0.14%	0.14%
letter v	96.40%	0.00%	3.60%
letter w	84.09%	0.28%	15.63%
letter x	100%	0.00%	0.00%
letter y	99.28%	0.24%	0.48%
letter z	90.32%	0.00%	9.68%

TABLE I

PERCENTAGE OF CORRECTLY AND INCORRECTLY RECOGNIZED BOUNDING BOXES AS WELL AS THE NUMBER OF ANOMALIES FOUND.

B. Tesseract errors

We classified the bounding boxes extracted by Tesseract into three categories:

1) Correct: the bounding boxes that were correctly recognized as example in Fig. 3.a.
2) Incorrect: the bounding boxes that were incorrectly recognized by Tesseract as examples in Fig. 3.b-f that were recognized as letter 'e'.
3) Anomaly: the bounding boxes that contain specific symbols or cropped characters (see Fig. 3.g-h).

The number of correctly and incorrectly recognized bounding boxes as well as the number of anomalies found in bounding boxes are shown in Table I. We note that more than a half of the characters have both incorrect recognition and anomalies in bounding boxes. The anomalies are presented by noise points added by double P&S process or by inaccurate bounding box extraction due to character overlapping. In the rest of our experiments only the correctly extracted bounding boxes are used.

C. Comparison of the two approaches

We compare the character recognition results using Tesseract and PCA based recognition system without and with preprocessing stage. In experiments with Tesseract 3.05 with the default English training, we only count the number of characters that were incorrectly recognized, without taking into account anomalies (such as detection of specific symbols or uppercase characters as lowercase characters, or the use of cropped bounding boxes in the character recognition). For the PCA based approach, the bounding boxes were resized to a 50×50 pixel size. For training, we used 26 bounding boxes (lowercase characters) extracted from a digital document image.

	Tesseract	Tesseract with pre-proc.	PCA 50×50	PCA 50×50 with pre-proc.
letter a	2.64%	0.09%	0.00%	0.00%
letter b	1.50%	0.90%	0.00%	0.00%
letter c	2.20%	0.60%	0.70%	0.70%
letter d	0.00%	0.00%	0.00%	0.00%
letter e	0.52%	0.18%	0.00%	0.13%
letter f	7.54%	0.93%	0.00%	0.00%
letter g	0.00%	0.00%	0.00%	0.00%
letter h	0.09%	0.55%	0.00%	0.00%
letter i	0.91%	0.28%	3.60%	1.37%
letter j	23.08%	8.33%	0.00%	0.00%
letter k	0.00%	0.00%	0.00%	0.00%
letter l	25.83%	63.55%	20.27%	13.51%
letter m	0.00%	0.00%	0.00%	0.00%
letter n	0.32%	0.00%	0.00%	0.00%
letter o	5.12%	0.24%	0.00%	0.00%
letter p	0.20%	0.00%	0.00%	0.00%
letter q	6.25%	0.00%	0.00%	0.00%
letter r	0.44%	0.00%	0.00%	0.00%
letter s	0.96%	0.00%	0.00%	0.00%
letter t	0.77%	0.05%	0.00%	0.00%
letter u	1.82%	0.00%	0.00%	0.00%
letter v	0.42%	0.85%	0.00%	0.00%
letter w	0.00%	0.00%	0.00%	0.00%
letter x	0.00%	0.00%	0.00%	0.00%
letter y	0.00%	0.00%	0.00%	0.00%
letter z	0.00%	0.00%	0.00%	0.00%

TABLE II

PROBABILITY ERROR RATE FOR TESSERACT OCR ALGORITHM AND FOR PCA BASED APPROACH.

The recognition results for each lowercase letter are shown in Table II. The recognition results are evaluated using probability error rate, i.e. the percentage of characters that are incorrectly recognized. We note that the PCA approach has better recognition results for all characters except of the letter 'i'. There are also several errors in the recognition of the letters 'c' and 'l', but the probability error rate is smaller in both cases in comparison with Tesseract results. Additionally,

the use of the pre-processing step improves the recognition results for both approaches: the PCA approach has less errors in recognition of letters 'i' and 'l' and Tesseract improves the recognition of all letters except letter 'l' (which is often recognized as a letter 'i' or 't').

In order to show the performance of the PCA approach (without pre-processing stage) applied to samples printed and scanned at 300 dpi and 600 dpi as well as the samples printed and scanned twice at 600 dpi, we listed the error probability of incorrect detection in Table III. We note that the error rate for samples printed and scanned once is less than 1% and almost all errors have been done in the recognition of the letter 'i' and the letter 'l' (which are also the most badly detected letters in the Tesseract recognition results).

		With rejection threshold	
	PCA 50×50	PCA 50×50	PCA 80×80
P&S 300 dpi	0.004%	1.533%	0.630%
P&S 600 dpi	0.016%	1.362%	0.035%
double P&S 600 dpi	2.211%	11.954%	1.531%

TABLE III

ERROR RATE OF INCORRECT DETECTION USING THE PCA BASED APPROACH.

Several experiments have also been done with the same Arial font documents but with another font sizes (8 and 10 font sizes). The first results show that the documents printed and scanned with 300 dpi have an error rate between 1.99% and 4.81%, which is quite high. At the same time, the documents printed at 600 dpi do not have errors in character recognition. The number of documents with 8 and 10 font sizes was not sufficient to make a conclusion about the performance of PCA based recognition system, however, we can conclude that the size of images used in PCA classification is a very important parameter. Therefore, we need to do supplementary tests in order to define the optimal image size.

D. Rejection of unknown characters

The optional functionality of PCA classification is the use of a distance threshold to reject anomaly images (i.e the images that do not belong to the training data set). This threshold is set heuristically. In our experiments, the anomaly images are the badly centered bounding boxes or the erroneous bounding boxes (see the examples in Fig. 2.b-d and Fig. 3.g-h respectively).

The results of character recognition with rejection threshold are shown in Table IV. For this experiments we have used two image sizes: 50×50 pixels and 80×80 pixels. We can note that results with image size 50×50 are more worse than those obtained with image size 80×80. In fact, when the size of images is small, it is not easy to choose a good rejection threshold. In the experiments, the bad bounding boxes (as illustrated in Fig. 2 and Fig. 3) are always rejected thanks to the threshold. Nevertheless, some correct characters are also rejected due to the crucial impact of double P&S process.

The comparison of Tesseract character recognition and PCA based recognition system (without pre-processing step) results with image size 80×80 pixels with rejection threshold

		With rejection threshold	
	Tesseract	PCA 50×50	PCA 80×80
letter a	2.64%	0.18%	0.00%
letter b	1.50%	0.00%	0.00%
letter c	2.20%	0.90%	12.41%
letter d	0.00%	0.31%	0.00%
letter e	0.52%	0.42%	0.13%
letter f	7.54%	0.22%	0.00%
letter g	0.00%	0.56%	0.00%
letter h	0.09%	0.54%	0.73%
letter i	0.91%	64.65%	1.26%
letter j	23.08%	7.69%	0.00%
letter k	0.00%	0.00%	0.00%
letter l	25.83%	91.07%	5.03%
letter m	0.00%	1.04%	0.00%
letter n	0.32%	0.54%	0.11%
letter o	5.12%	0.19%	0.00%
letter p	0.20%	0.00%	0.00%
letter q	6.25%	0.00%	0.00%
letter r	0.44%	0.82%	0.00%
letter s	0.96%	0.24%	0.00%
letter t	0.77%	0.23%	0.00%
letter u	1.82%	0.36%	0.00%
letter v	0.42%	0.00%	0.00%
letter w	0.00%	0.62%	0.00%
letter x	0.00%	0.00%	0.00%
letter y	0.00%	0.00%	0.00%
letter z	0.00%	0.00%	0.00%

TABLE IV

PROBABILITY ERROR RATE FOR TESSERACT OCR ALGORITHM AND THE PCA BASED APPROACH.

are shown in Fig. 6. The PCA based method performs better for all characters except for three: 'c', 'h', 'i'. The letter 'c' is often detected as 'o', while the letters 'h' and 'i' have the distances bigger that the fixed rejection threshold. We suppose that we can improve these results with some pre-processing steps that could improve the sharpness of character contours.

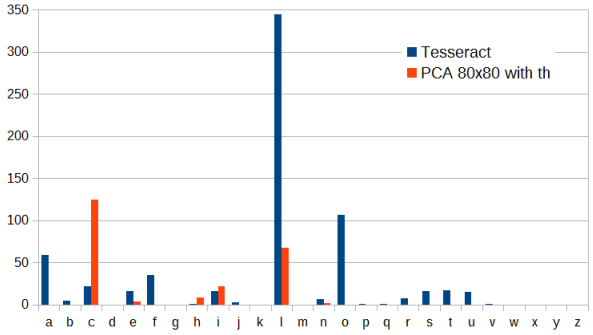

Fig. 6. The comparison of number of incorrectly recognized characters using Tesseract 3.05 and the PCA based approach.

E. Pros and cons of PCA based recognition

The PCA based method has its pros and cons. We need to do the supplementary experiments in order to find the optimal image size and the threshold value. Additionally, both methods were tested with only one printer-scanner pair, therefore the supplementary tests need to be done with different printer-scanner pairs. However, the PCA based approach has some strong points:

- It outperforms the Tesseract OCR technique when character recognition is done after double P&S process (see Section V-C).
- It can use morphological filters for the correction of P&S distortions. These filters could fill the halftone imperfection in the characters and, thus, slightly improve the recognition results.
- It can use grayscale images for character recognition. The binarization methods can remove important parts of characters and decrease the recognition results.
- It can reject the unknown or bad defined bounding boxes. This property is important as the errors of bounding box extraction can strongly change the character recognition results (see Section V-D).
- It does not use a dictionary for character recognition. This point is very important while talking about document authentication as we do not need that the word recognition corrects the errors produced by an opponent in faked documents.

In the same time the PCA based recognition system has several drawbacks:

- Sensitivity to the bounding box size. The recognition results can be changed in function of the bounding box size.
- Sensitivity to bounding box quality, i.e. the additional background lines/columns or a not centered character in bounding box can crucially change the recognition results.
- Sensitivity to printer and scanner used for the rejection threshold determination. Due to a specific signature of each printer and scanner, the determination of an universal rejection threshold seems to be a difficult task.

VI. Conclusions and future work

The integrity check of printed and scanned documents is a hot topic these days. One of the evident solutions is the use of OCR and cryptographic hash for text authentication. This approach works well with digital documents, but is not trivial when the document is printed and scanned once or twice. The top OCR algorithms have a quite good performance when the document is printed and scanned once, but these solutions do not work for documents printed and scanned twice.

In this paper we show the problems of the Tesseract OCR when it deals with documents printed and scanned twice. The main problem is the lack of a pre-processing stage before character recognition that leads to incorrect bounding box extraction and, thus, incorrect character recognition.

We suggest to use the PCA classification with a pre-processing stage in order to improve the character recognition in double printed and scanned documents. The experimental results show that the proposed solution gives better recognition results in comparison with Tesseract 3.05. The proposed pre-processing stage improves the results of Tesseract recognition as well. Furthermore, it is possible to determine the threshold in order to reject the badly extracted or anomaly bounding boxes. Nevertheless, the PCA based method is sensitive to bounding box extraction and to P&S noise.

In future we can imagine several paths in order to thoroughly study the impact of double P&S process to text documents. The first path is to evaluate the PCA based method using a big amount of printer-scanner pairs in order to determine the common features and differences of the impact after a double P&S process. That will help us to explore the second path which is the use of additional pre-processing steps (such as denoising, sharpening of character contours) to improve the character structure after double P&S process. The third path is the search of optimal image size (which is insensitive to font type and font size) for PCA classification that has to be found experimentally. Finally, we need to find the optimal rejection threshold that can be used for any font, any font size and any pair of printer-scanner used.

Acknowledgment

This work has been financed by the French National Research Agency (ANR) project SHADES referenced under ANR-14-CE28-0022.

References

[1] C. Baras and F. Cayre. 2D bar-codes for authentication: A security approach. In *Proceedings of European Signal Processing Conference (EUSIPCO)*, pages 1760–1766, 2012.

[2] C. M Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[3] S. Eskenazi, P. Gomez-Krämer, and J-M Ogier. When document security brings new challenges to document analysis. In *Proceedings of Computational Forensics*, pages 104–116. Springer, 2015.

[4] A. Ferreira, L. C Navarro, G. Pinheiro, J. A dos Santos, and A. Rocha. Laser printer attribution: Exploring new features and beyond. *Forensic science international*, 247:105–125, 2015.

[5] A. T. P. Ho, B. A. M. Hoang, W. Sawaya, and P. Bas. Document authentication using graphical codes: Reliable performance analysis and channel optimization. *EURASIP Journal on Information Security*, 2014(1):9, 2014.

[6] A. TS Ho and F. Shu. A print-and-scan resilient digital watermark for card authentication. In *Proceedings of Information, Communications and Signal Processing*, volume 2, pages 1149–1152. IEEE, 2003.

[7] 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. In *Security, Steganography, and Watermarking of Multimedia Contents*, pages 430–440, 2005.

[8] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[9] R. Pizzio. Hardcopy watermarking for document authentication. In *Watermarking-Volume 1*. InTech, 2012.

[10] S. Shang, N. Memon, and X. Kong. Detecting documents forged by printing and copying. *EURASIP Journal on Advances in Signal Processing*, 2014(1):140, 2014.

[11] A. Smith. Identity fraud: A study. In *Technical Report July, Great Britain Economic and Domestic Secretariat*, 2002.

[12] R. W. Smith. History of the Tesseract OCR engine: what worked and what didn't. In *Proceedings of SPIE, Document Recognition and Retrieval XX*, volume 8658, pages 865802–865812, 2013.

[13] L. Tan and X. Sun. Robust text hashing for content-based document authentication. *Information Technology Journal*, 10(8):1608–1613, 2011.

[14] L. Tan, X. Sun, Z. Zhou, and W. Zhang. Perceptual text image hashing based on shape recognition. *Advances in Information Sciences and Service Sciences (AISS)*, 3(8):1–7, 2011.

[15] R. Villán, S. Voloshynovskiy, O. Koval, F. Deguillaume, and T. Pun. Tamper-proofing of electronic and printed text documents via robust hashing and data-hiding. In *Proceedings of Electronic Imaging 2007, International Society for Optics and Photonics*, pages 65051T–65051T, 2007.

[16] L. Yu, X. Niu, and S. Sun. Print-and-scan model and the watermarking countermeasure. *Image and Vision Computing*, 23(9):807–814, 2005.