ASSESSING THE OCR DEGRADATION IN THE GENERATION OF JPEG, PNG, AND TIFF FILES FROM ADOBE PDF

Gabriel de França Pereira e Silva, Rafael Dueire Lins,

Departamento de Eletrônica e Sistemas - Universidade Federal de Pernambuco – Brazil
gfps@cin.ufpe.br rdl@ufpe.br

***Abstract*—Adobe Portable Document Format is de facto standard today due to its widespread use. One of the features of Adobe pdf is that it allows exporting documents as images that may be saved in JPEG, PNG, and TIFF formats. This paper uses an OCR platform to quantitatively assess the quality of these image file formats.**

Keywords- JPEG; PNG; TIFF; OCR; quality

I. INTRODUCTION

Document "understanding" is fundamental to fully meet the short, medium and long terms aims of digital libraries [1][2]. Such understanding goes far beyond the document content itself and encompasses the way it is generated and the possibilities of converting between digital formats. The Postscript format was a milestone in the history of file printing. It pioneered a device independent printing format and established and intermediate language for device drivers. Although it was increasingly popular in the late 1980s its importance suddenly faded. The role Postscript aimed to play is today performed by Adobe® PDF. The Portable Document File, pdf – for short, is of widespread use today. *.pdf files are generally used for documents but it is also adopted by some Microsoft Office© Tools, such as Powerpoint©, as a way to output files one may publish but one does not want it to be (easily) altered. PDF files have tags which delimiter each kind of embedded file. Curiously, the authors observed that an embedded proprietary file gets replaced by a "similar" public domain one. For instance, an embedded *.gif file gets replaced by a *.png file with (possibly) the same information. One of the possibilities Adobe pdf is to generate image files. In theory, the Adobe Acrobat 6.0 Professional exports texts into five different image files: BMP [3] [16], JPEG [2][4][8][13], PNG [3][4][7][10], TIFF [3][17] and JPEG2000 [12][14][15]. However, the JPEG2000 files generated by the authors were not compatible with any other software available, including ImageJ, the ABBY Finereader®[18], or PaintShop Pro®. The technical literature reports nothing about the features of the image files generated from pdf. Table 01 shows the resolution of the image files obtained from pdf. This paper assesses the quality of the images bmp, jpg, tiff, and png generated from Adobe Acrobat 6.0 Professional in two different ways. The first of them is via OCR. The image files are transcribed by using the ABBYY FineReader [18] and compared with the original text. The second quantitative evaluation of the quality of the image generated is by calculating the PSNR (peak signal-to-noise ratio) between each image file and its bmp version, which was generated using the ABBYY FineReader and had its resolution decreased to be compatible with the one generated by the Adobe pdf. This way one has an account of the quality of the images generated.

File Format	Resolution
JPEG	1190x1683
PNG	1190x1683
TIFF	1190x1683

Table 1. Resolution of different file formats.

II. BMP, JPEG, PNG, AND TIFF FORMATS

This section briefly explains the main features of the file formats target of the pdf file and points at further references for details.

A. BMP

The bmp [3][16] file was created by Microsoft and IBM and is therefore very strictly bound to the architecture of the main hardware platform that both companies support: the IBM compatible pc. All values stored in the bmp file are in the Intel format, sometimes also called the little endian format because of the byte order that an Intel processor uses internally to store values. The *.bmp files are the way, Windows stores bit mapped images. The image data is bit packed but every line must end on a dword boundary - if that is not the case, it must be padded with zeroes. BMP files are stored bottom-up, which means that the first scan line is the bottom line. It has four "incarnations", two under Windows (new and old) and two under

OS/2. BMP images can range from black and white (1 bit per pixel) up to 24 bit color (16.7 million colors). While the images can be compressed, this is rarely used in practice.

B. JPEG

The JPEG (Joint Photographic Experts Group) [3][4][8][13] is the result of a team effort of members of ISO (International Standards Organization) and ITU-T (International Telecommunication Union), whose official name is ISO/IEC JTC1 SC29 Working Group 1. JFIF is the most used file format that implements JPEG. However, in general most people refer only to JPEG. JPEG compression scheme is quite ingenious employs transformation of the color space and other artifices. There are four different methods for JPEG:

- Without loss encoding: This option is not efficient, it is not completely clear that there is no real loss, thus its usage is not recommended (overall in medical imaging).
- Sequential encoding: each image component is encoded in the same order as scanned (left to right, top to bottom);
- Progressive: the image is encoded in several steps
- Hierarchical: the image is encoded in different resolutions. Low resolution versions of the image may be visualized without uncompressing the whole image.

The JPEG without loss algorithm (a) uses a predictive followed by statistical encoding of the image. The versions of JPEG with losses (b, c and d) make use of the Discrete Cosine Transform in blocks of 8x8 pixels, followed either by Huffman or Arithmetic encoding of coefficients [5][6]. Progressive JPEG encoding makes use of a buffer in the output of the DCT encoder to "reorganize" coefficients before storage. Different "chunks" of coefficients are sent at each encoding step or scan. The user may set the number of scans required. References [4][8][13] bring details of JPEG different features.

C. PNG

The Portable Network Graphics format was developed in 1995 in alternative to GIF [11], to avoid UNISYS patent on LZW compression algorithm. PNG file format was meant to be simple, portable, easily extensible, open code and free. PNG is better than GIF for allowing up to 48 bits for color representation (GIF allows only 8 bits for color information). PNG compression algorithm is based on the Deflate, created by Phil Katz, which is based on LZ77 with a sliding window and ordered hash table [5][6]. Reference [19] points out that, amongst the lossless algorithms studied, PNG presented the best performance for all kinds of images and resolutions analyzed.

D. TIFF

The Tagged Image File Format (TIFF) [3][17] is a file format for storing images, including photographs and line art. It is now under the control of Adobe. Originally created by the company Aldus for use with what was then called "desktop publishing", the TIFF format is widely supported by image-manipulation applications, by publishing and page layout applications, by scanning, faxing, word processing, optical character recognition and other applications. Adobe Systems, which acquired Aldus, now holds the copyright to the TIFF specification. TIFF has not had a major update since 1992, though several Aldus/Adobe technical notes have been published with minor extensions to the format, and several specifications, including TIFF/EP, have been based on the TIFF 6.0 specification.

III. TEST IMAGES AND METHODOLOGY

The test images used in this work were obtained from the proceedings of the Brazilian Telecommunications Symposium from 2000 and 2005, with 782 and 1,358 pdf pages, respectively. An example of page may be found in Figure 01.

File Format	Average size
JPEG	875 KB
PNG	779 KB
TIFF	783 KB

Table 2. Average size for the different file formats (2,140 images).

As may be observed in Figure 01, the page includes graphical elements. Experiments performed showed that Adobe pdf was able to export embedded jpeg files directly without further image degradation. That means that somehow the original file was exported. The same phenomenon was not observed whenever neither the original file was not jpeg, nor when the target file format was not jpeg.

Two methods were used to measure the quality of the images:

1. Analysis by PSNR, for this we used the full area of the image, i.e. non-textual elements were incorporated into the analysis (see red blocks into Figure 1).

2. Analysis by Automatic Transcription (OCR) - During the OCR images only areas consist of text have been transcribed by the tool ABBYY FineReader 9.0, the areas belonging to non-textual elements were discarded manually (by interface tool).

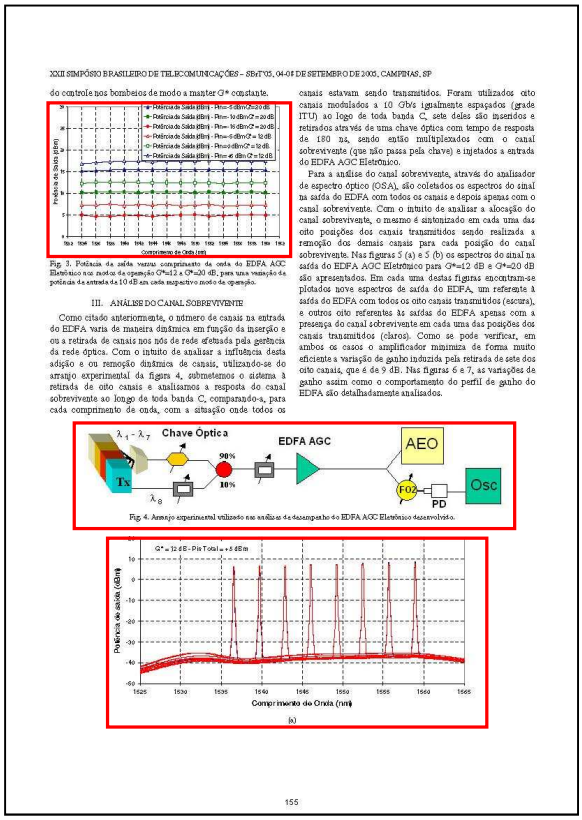

Figure 1. Page from SBRT 2005.

Table 03 shows the results the PSNR (peak signal-to-noise ratio) between the images that have the same resolution.

Original	Target	PSNR
JPEG	JPEG	1.0000
JPEG	TIFF	0.9642
JPEG	PNG	0.9642
TIFF	JPEG	0.9877
TIFF	TIFF	0.9913
TIFF	PNG	0.9913
PNG	JPEG	0.7915
PNG	TIFF	0.7909
PNG	PNG	0.7909

Table 3. PSNR of image exportation from pdf.

In order to compare the quality of the images generated by Adobe pdf an intermediate bmp image was generated using the ABBY FineReader, which may take as input a pdf file and generate an uncompressed bmp file of resolution 2,479x3,508 pixels. MATLAB was used to decrease the resolution of the BMP files in order to make them compatible with the resolution of the images generated by pdf to allow for the PSNR to be calculated. As the PSNR obtained between the tiff and png images and the bmp was exactly the same, the PSNR was re-calculated between TIFF and PNG images and curiously the two image formats yield exactly the same images. The results obtained for the 3,140 images analyzed are shown in Table 03. In general, from the PSNR perspective, one may say that the images generated from Adobe pdf are of very high quality and quite close to the BMP image generated by the ABBY FineReader.

File Format	Reference Format	PSNR
JPEG	BMP	0.9200
TIFF	BMP	0.9760
PNG	BMP	0.9760
TIFF	PNG	1.0000

Table 4. PSNR 2,140 document pages having a bmp and png images as reference.

Assessing image quality in general is extremely difficult. In the case of the experiments whose results are presented in Tables 02 and 03 the inputs for comparison are two images, thus there are chances of performing a fair assessment. How to assess the quality of an image generated as a raster from a “text” file? Analyzing the quality of the transcription of such image in comparison with the text of the original file may provide a clue about the quality of the text-to-image conversion. That is the approach followed in this paper. However, on its turn, analyzing the results of OCRs is far from being a trivial task. The methodology presented in reference [20] which takes into account the nature of the errors in transcription was adopted here. Only character errors were analyzed. The character errors are classified according to:

1. Character replacement;
2. Missing characters;
3. Character insertion;
4. Punctuation errors.

The results obtained for the 2,140 test images are shown in Table 05. If one divides the image resolution of the images generated by Adobe pdf by the size of the document page one finds a resolution close to 150 dpi, while in the case of the BMP image generated by the ABBY FineReader the resolution reaches the 300

dpi. The data in Table 05 shows that the results obtained in the transcription of the images are very close amongst themselves. The performance of JPG was slightly worse than PNG and TIFF, presenting a behavior consistent with the results of the PSNR obtained. Curiously, PNG and TIFF yielded a performance extremely similar, but not equal.

SBRT 2000	#Total characters 3,378,891			
	JPG	PNG	TIFF	BMP
replacement	840177	847464	847625	839001
punctuation	26659	26750	26720	26741
missing	129805	122443	121944	124984
insertion	223062	214575	214080	221568
SBRT 2005	#Total characters 5,107,207			
	JPG	PNG	TIFF	BMP
replacement	1344674	1341703	1341840	1341631
punctuation	90960	90608	90589	90912
missing	102148	96412	97616	98365
insertion	315967	310572	312267	311532

Tab. 5. Character errors found in PDF generated images

IV. CONCLUSIONS

This paper provides evidences that the generation of JPEG, PNG and TIFF, image files from Adobe pdf files using Adobe Acrobat Professional 6.0 yields very good quality images. The TIFF and PNG images seem to be equal, modulo changes in their headings. JPG as a file format with losses performed very well in general, but slightly worse than PNG and TIFF. The intuition that would make one believe that the higher the PSNR of an image, the better the transcription obtained, was proved correct herein.

REFERENCES

[1] H.S.Baird. Document image defect models and their uses. ICDAR 1993, pp. 62-67, 1993.

[2] M.Cheriet and R.F.Moghaddam. DIAR: Advances in Degradation Modeling and Processing, ICIAR 2008, LNCS(5112):1-10, Springer Verlag, 2008.

[3] J. da Silva; *et al.* "A New and Efficient Algorithm to Binarize Document Images Removing Back-to-Front Interference". J.Universal Computer Science, v(14):299-313, 2008.

[4] R. D. Lins, *et al.* "An Environment for Processing Images of Historical Documents. Microprocessing & Microprogramming", v(40):939-942, N-Holland, 1993.

[5] G.Sharma, "Show-through cancellation in scans of duplex printed documents", IEEE Trans.I.Proc, v10(5):736-754, 2001.

[6] R.D.Lins, *et al*. "Detailing a Quantitative Method for Assessing Algs. to Remove Back-to-Front Interf. in Docs". J.Universal Computer Science, v. 14, p. 266-283, 2008.

[7] G.Meng *et al*. Circular Noises Removal from Scanned Doc. Images. ICDAR 2007, pp. 183-187, IEEE Press, 2007.

[8] D.Möri and H.Bunke. Automatic interpretation and execution of manual corrections on text documents. Hbook Char. Recog. Doc. Image Analysis, pp 679-702. World Scientific, 1997.

[9] J.Stevens, A.Gee, C.Dance. Automatic processing of document annotations. British M.Vision Conf., v(2): 438-448, 1998.

[10] J.K.Guo and M.Y.Ma. Separating handwritten material from machine printed text using hidden markov models. ICDAR 2001, pp.436-443, 2001.

[11] Y.Zheng, H.Li, and D.Doermann. The segmentation and identification of handwriting in noisy document images. DAS02, LNCS 2423, pp.95-105, Springer Verlag, 2002.

[12] T.Nakai, K.Kise, M.Iwamura. A method of annotation extraction from paper documents using alignment based on local etc, ICDAR 2007, pp.23-27, IEEE Press, 2007.

[13] J.R.Caldas Pinto *et al*. Underline Removal on Old Documents. ICIAR 2004, LNCS(3212), v(2):226-233, 2004.

[14] V.Bruni, P.Ferrara, and D. Vitulano. Color Scratches Removal Using Human Perception, LNCS(5112):33-42, S.Verlag, 2008.

[15] M.Wirth and B.Bobier. Supression of Noise in Historical Photographs Using a Fuzzy Truncated-Median Filter. ICIAR 2007, LNCS(4633):1206-1216, Springer Verlag. 2007.

[16] K.C.Fan, Y.K.Wang, T.R.Lay, Marginal noise removal of document images, Patt.Recog. 35, 2593-2611, 2002.

[17] D.X.Le, Automated borders detection and adaptive segment. for binary document images. National Library of Medicine. http://archive.nlm.nih.gov/pubs/le/twocols/twocols.php

[18] B.T.Ávila and R.D.Lins, A New Algorithm for Removing Noisy Borders from Monochromatic Documents, ACM-SAC'2004, pp 1219-1225, ACM Press, March, 2004.

[19] B.T.Ávila and R.D.Lins, Efficient Removal of Noisy Borders from Monoch.Docs, LNCS(3212):249-256, S.Verlag, 2004.

[20] R.Gomes e Silva and R.D.Lins. Background Removal of Document Images Acquired Using Portable Digital Cameras. LNCS(3656): 278-285, Springer Verlag 2005.