Generation of Images of Historical Documents by Composition

Carlos A.B. Mello

Faculdade Santa Maria, Recife (PE), Brazil

cabm@netpe.com.br

Rafael D. Lins

Departamento de Eletrônica e Sistemas, UFPE, Recife (PE), Brazil

rdl@ee.ufpe.br

Abstract

This paper describes a system for efficient storage, indexing and network transmission of of historical documents. First, documents are decomposed into their features such as paper texture, colours, typewritten parts, handwritten parts, pictures, etc. Common components are factores out. Document retrieval forces the re-assembling of the document, synthetising an image visually close to the original document. The information needed to build the final image occupies, in average, 2 Kbytes performing a very efficient compression scheme.

1 Introduction

The work reported herein is part of the Nabuco Project[5][10] for preservation and broadcasting of the letters and documents from Joaquim Nabuco[1]'s bequest. The file is composed of almost 6,500 documents from the end of the nineteenth century, totalizing over 30,000 pages.

The Nabuco Project is developed by the Federal University of Pernambuco jointly with the Joaquim Nabuco Foundation (a social science research centre), both in Recife (Brazil). Documents are digitized in true colour with 200 dpi resolution and stored in JPEG [9] file format with 1% loss. Even in this format each image of a document reaches, in average, 380 Kb.

A processing environment was envisaged to extract the basic features of documents, which allows for later image re-assembling. The extraction of documents is performed by the blocks presented in Figure 1.

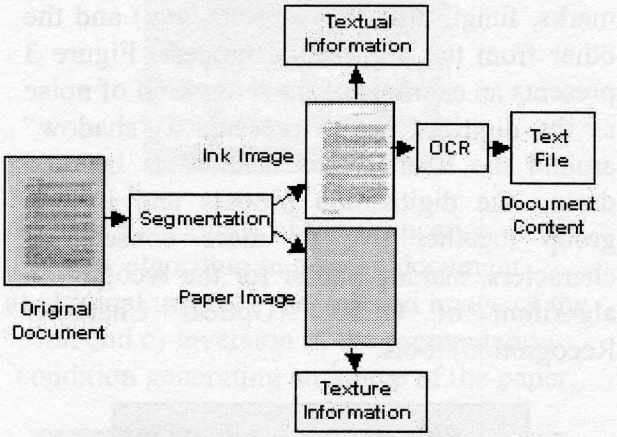


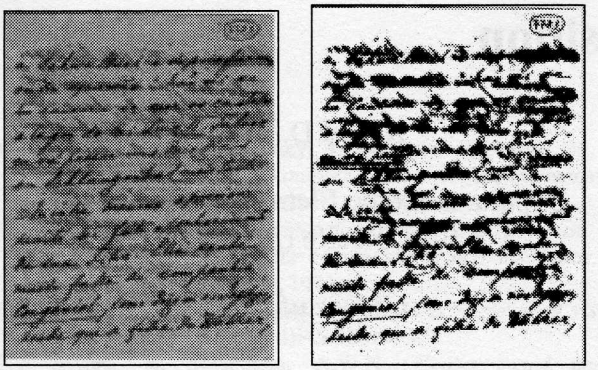
Figure 1. Block diagram for the data extraction of the paper and text image.

In order to obtain satisfactory results, several new algorithms were developed within the scope of the project and are described herein.

Ink and paper segmentation is not always a simple task in this kind of image. In some documents, the ink has faded; some of the others were written on both sides of the paper and the ink transposed the document presenting back-to-front interference. A conversion into a monochromatic version of this kind of documents using a nearest colour threshold algorithm [3] does not achieve high quality results as can be seen in Figure 2

[1] Brazilian statesman, writer, and diplomat, one of the key figures in the campaign for freeing black slaves in Brazil, Brazilian ambassador to London (b.1861-d.1910)

below. A new entropy-based segmentation algorithm presented in [6][7] yielded better results.



a) b)

Figure 2. Sample document of the Nabuco's bequest where the paper is written on both sides. a) Original document in greyscale and b) its conversion to monochromatic using a nearest colour algorithm.

The images present two kinds of noise: one from the paper itself (such as humidity, marks, fungi, dirt, finger prints, etc.) and the other from the digitization process. Figure 3 presents an example of the latter kind of noise as the digitized image presents a "shadow" around the letters. This shadow is inserted during the digitization process and it may group together two or more consecutive characters, making harder for the recognition algorithm of OCR's (Optical Character Recognition) tools.

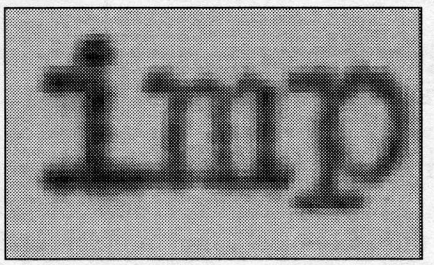


Figure 3. "Shadow" around characters due to the digitization process.

After segmentation, the paper and the ink images are processed separately for extracting their main features.

The ink part is processed by an OCR tool to generate an annotated text with entries to a fontset database. As it needs the use of an OCR, this system is applied only to typewritten documents by now.

Document re-assembling follows the scheme described in Figure 4 below. The texture information is used to build a blank sheet of paper visually similar to the original one.

On the other hand, the "text file" (output of OCR) generates a text image, using some default settings for the document (as margins, blank space size, space between lines, etc) and a database of images of characters. Some other data ("textual information") such as colour, hues, etc, extracted from the original file are used to colour the "textual image". The coloured textual image is added on top of the "blank sheet of paper" yielding the sinthetic image.

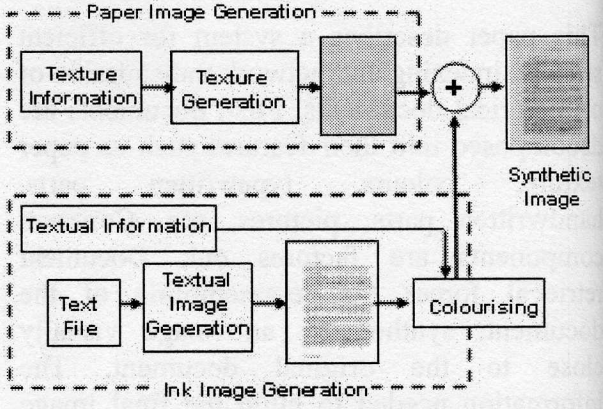


Figura 4. Scheme for generating the synthetic image of a historical document.

The information needed to build the final image occupies less than 2 Kbytes.

The specification and analisys of a system that generates a synthetic image of typed documents is the main goal of this paper.

Next, each main step of the generation process is analysed beginning with a review of the entropy-based segmentation algorithm followed by the algorithm for text and texture synthesis.

2 The Entropy-Based Segmentation Algorihtm

The efficient segmentation between ink and paper is fundamental to the document processing environment presented herein. For this purpose, entropy-based segmentation

algorithms presented the best results when applied to a set of more than 200 documents.

For greyscale images, at first, it is found the most frequent colour, *t*. As the environment works with images of letters and documents, it is reasonable to suppose that this colour belongs to the paper. The entropy [1] of the pixels below and above this value (*Hb* and *Hw*, respectively) is evaluated:

$$Hb = -\sum_{i=0}^{t} p[i]\log(p[i]) \qquad Hw = -\sum_{i=t+1}^{255} p[i]\log(p[i])$$

where the logarithmic basis is taken as the product of the dimensions of the image (height *versus* width) and *p[i]* is the probability of the colour *i* is present in the image. The entropy of the complete histogram, *H*, is evaluated as the sum of *Hw* and *Hb* and it defines two multiplicative factors, *mw* and *mb*, experimentally determined by:

- If H≤0.25, then mw=2 and mb=3;
- If 0.25<H<0.30, then mw=1 and mb=2.6;
- If H≥0.30, then mw=mb=0.8.

These values of mw and mb were applied to a set of 500 images achieving very satisfactory results.

The image is re-built with pixel *i* with colour *colour[i]* converted to *white* if:

$$(\text{colour}[i]/256) \geq (mw.Hw + mb.Hb)$$

else it remains the same (generating a new greyscale image) or it is converted to *black* (to generate a monochromatic image). This is called the *segmentation condition*. An example of the application of the algorithm may be found in Figure 5. An inversion of this condition generates an image where the pixels classified as ink are turned into white remaining only the paper texture (Figure 5.c).

One can see *Hw* and *Hb* as projections of the entropy itself. The value of *mw.Hw* + *mb.Hb* in the segmentation condition causes a change in entropy weighted by these projections.

For true colour images, the algorithm works as above but now it is applied for each of the RGB (*Red*, *Green* and *Blue*) components. A pixel is classified as paper (thus turned into white) if the segmentation condition results true for, at least, one of the components R, G or B. Otherwise, its colour remains unchanged. Figure 6 shows an example of a true colour image with back-to-front interference and the result after the application of the algorithm.

The use of the algorithm increased the response of OCR tools more than 10%. One of the cases where the algorithm did not work well was in very faded documents. In these cases, re-tunning of the image brightness and contrast is necessary before applying the algorithm. In other cases, it was necessary to apply the algorithm in cascade to achieve better results.

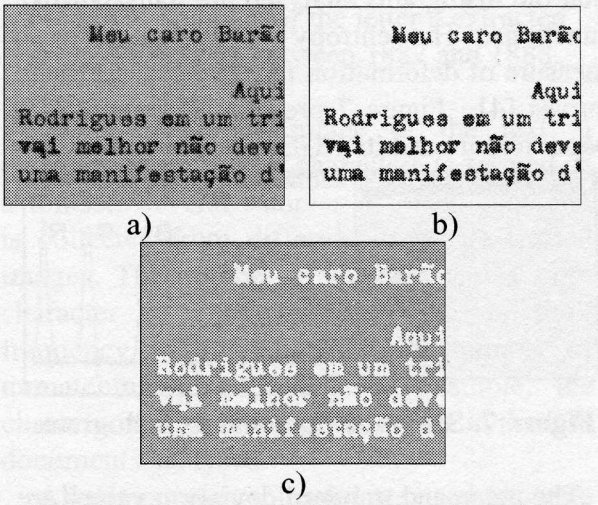


Figure 5. Example of the application of the new algorithm in a typed document.
a) Original image, b) segmented image of the ink and c) inversion of the segmentation condition generating an image of the paper.

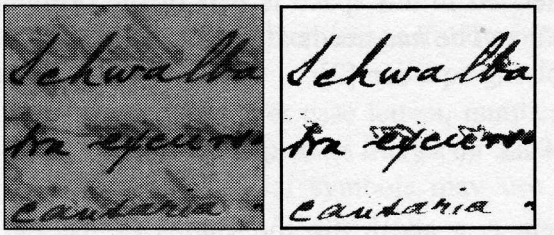


Figure 6. New entropy-based segmentation algorithm applied to an true colour image with back-to-front interference.
a) Original image and b) segmented version.

Both images produced by the algorithm are used in the next steps of the system for automatic generation, as later detailed. From now on, the segmented image of pixels that were classified as paper is called *texture*

image and its negative (the pixels classified as ink) is called *textual image*.

Detailed information on the segmentation algorithm may be found in [6][7].

3 Texture Synthesis

Samples of the background of 200 colour images were collected and analysed. From them the mean, standard deviation and entropy of the histograms of RGB components were extracted. Figure 7 shows that the histograms analysed are gaussian-like functions. The entropy is evaluated as a measure of deformation from perfect gaussian curves [4]. Figure 7 presents an example of paper texture and its RGB histogram.

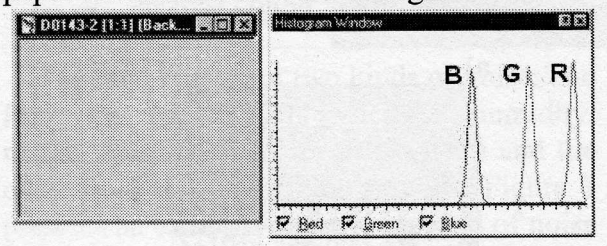


Figure 7. Sample texture and its histogram.

The mean and standard deviation values are used to specify the colour space of the image which has values between *mean – standard_deviation* and *mean + standard_deviation* for each of the three components. However, not all colours generated in this space is part of the original texture. The *hue* needs also to be analysed by applying equation [2]:

$$hue = \cos^{-1}(((r-g)+(r-b))/2)/\sqrt{(r-g)^2+(r-b)*(g-b)}$$

where r, g and b are the values of the red, green and blue components for a given colour. For all 200 images analysed, it was observed that there is a predominant hue value (called *hue_max*). Some of these values are found in over 40% of an image. A new image is then created where each pixel has its colour determined by a hue value, no longer by its RGB components alone. The hue_max of this new image is evaluated and this value is the factor that defines if a colour formed by some triple RGB can be accepted or not in the synthetic image. The most frequent hue value works as a boundary factor for the possible colours that can be generated in the texture image.

The entropy of the hues (*h_hue*) in the images is also evaluated. As before, the entropy is evaluated using as logarithmic basis the product of the dimensions of the images. The need for h_hue is explained below.

The values of the mean and standard deviation of the histograms of the RGB components, hue_max and h_hue are all the information needed to create the synthetic image of the texture of the paper. The dimensions of the image are also stored totalizing 40 bytes only.

The histogram of the synthetic image is now created using the mean and standard deviation of the histograms of the original image. The gaussian-like functions for the RGB space is expanded by the RGB values bounded by *mean ± factor*variance* for each colour tone. The *factor* variable is defined by an interactive process where a gaussian function is created for the histogram using the mean and standard deviation of each of the RGB components. *factor* reaches its final value whenever the somatory of the amplitudes of the histogram is either greater or equal to the number of pixels of the image or it is 20% of this number (in this case a correction is done to complete the number of pixels). The use of the variance instead of the standard deviation "stretches" the amplitude of the mean value of the gaussian distribution.

The entropy of the hues is used to determine how the distribution approaches to a gaussian one. The system may use it to increase the accepted values of hues generated from the RGB space defined. A triple (r,g,b) from this space is added to a colour table and it can be in the final image if its hue value is between *hue_max – delta* and *hue_max + delta*, where delta = 10, if h_hue > 0.17 and delta = 1, otherwise. These values were found empirically and are called the *hue space*.

Parameter h_hue is also used to define how many times the RGB space is met, limited by the hue space. The entropy h_hue and the number of times hue_max may be found in a texture suggest that there is a relationship between them. A first-order function is evaluated so that one can define the percentage of hue_max (*%_of_hue_max*) in an image based on h_hue:

%_of_max_hue = 152.64*h_hue + 16.0089

Functions of higher orders were tested (second, third, fourth and fifth order) with no satisfactory results.

Whenever the number of colours that have the accepted values of hues reaches its maximum (defined by %_of_max_hue), the rest of the image is filled with colours from the RGB space even in case they do not belong to the hue space.

The colour of each pixel is determined by pseudo-random searches in the colour table until %_of_max_hue is reached and then in the RGB space. The synthesis of the sample texture of Figure 7 is presented in Figure 8.

The synthetic textures were analysed qualitatively by visual inspection and quantitatively by measuring the *Peak Signal-to-Noise Ratio* (PSNR) and *Analysis of Variance* (ANOVA). In both cases, a set of 200 textures were generated and compared achieving satisfactory results.

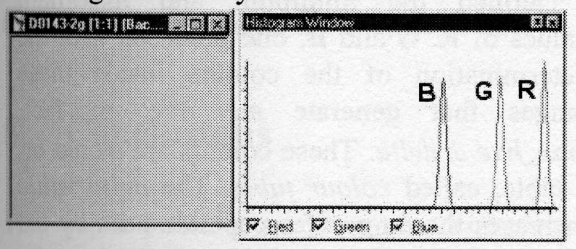


Figure 8. Automatic texture generation of sample texture of Figure 7.

4 Text Synthesis

The typed letters and documents of the bequest have very singular features. Most of the documents were written in the same typewriter, which did not allow for changes in the fontset. At the end of the 19th century, it was usual to writers and diplomats to take their own typewriters along in travels, similarly to carrying a laptop today. Figure 9 below shows four samples of the letter *u* extracted from four different documents between 1882 and 1888. One can observe that there is a strong fading in the upper right corner of the character presented in the four samples, suggesting that they were typed by the same machine.

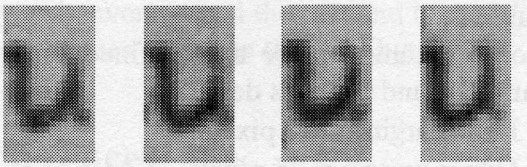


Figure 9. Samples of the letter *u* extracted from documents between 1882 and 1888.

To re-create the image of the text, a database of images of characters is formed as a fontset. Several samples of each character is collected from different greyscale textual images. The number of samples varies from character to character according to their frequency in the language, the number of unmatching pixels, etc. For example, the character "%" was found only once in the 200 documents analysed.

All the samples collected are used to create a unique fontset of images of each character. Each sample of a character has the same box and each pixel is compared looking for a majority vote or the mean of the pixels to define the colour of the final image of the character.

Each sample character image follows the rules:

- The upper and lower case letters, numbers and symbols are boxed in frames of 22x31 pixels. Some special symbols may use a larger box (as "%", "("and ")");
- Characters are stored in greyscale;
- Lower case letters are divided into three groups: wide (m and w), vertical (g, p, q, y and ç) and ordinary (the rest of the alphabet). According to this classification, the ordinary letters are layed in their frames with a 3-pixel left spacement and a 6-pixel bottom spacement; the wide letters have no extra space sideways; and the

vertical letters have only a 1-pixel margin to the bottom;

- It was chosen to store accentuated letters (such as à, á, â, etc.) instead of only its accents for performing optimization;
- Any symbol found in the text file that is not present in the character image database is exchanged for an asterisk.

Examples of some characters may be found in Figure 10.

Some parameters are also defined for the documents and taken as default:

- Left margin = 150 pixels
- Upper margin = 75 pixels;
- Blank space = 24 pixels;
- Space between lines = 70 pixels;
- Tabulation = 4 blank spaces.

The space between lines can have its value varied depending on the number of characters in the document. Some documents have almost the complete paper page typed and thus the space between lines is narrower (40 pixels now).

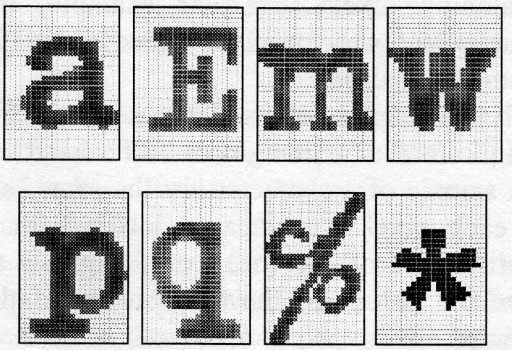


Figure 10. Sample character images extracted from typed documents of Nabuco's bequest.

In order to have the re-generation of the image of the document, the textual image created by the segmentation algorithm is inserted in an OCR tool and the resulting text file in annotated ASCII code is stored. At the present only one fontset was defined, with one image per ASCII symbol. We forsee the generation of a much larger fontset in which several images may be associated with each ASCII symbol. Looking up to the annotated ASCII code for a document in the fontset database one may regenerate an image with the textual part of the document in greyscale. A system to colourise this image is explained below. The system presented herein uses Omnipage which, in previous tests [8], presented the best results for our application.

The system does not deal with recognition errors yet. The textual image is generated directly from the OCR output.

4.1 Colourising the Text

Similarly to the texture generation, a 40 byte long binary data file is created with the same information from the original true colour textual image as before: mean and standard deviation for each RGB component, the most frequent hue value and the entropy of hues.

The range of RGB colours is then defined by its minimum and maximum values as follows:

$$max = mean + factor1 * std + factor2$$

and

$$min = mean - factor1 * std - factor2$$

where *mean* and *std* are the mean and standard deviation of the tones (R, G or B) in the original image and *factor1* and *factor2* are two constants defined by the following rule: if $h\_hue > 1$ then $factor1 = 1$, $factor2 = 0$ and $delta = 0$, else $factor1 = 2$, $factor2 = 1$ and $delta = 2$. Constants *factor1* and *factor2* may expand the colour space, while *delta* allows for a variation on value of *max_hue*.

Defined the minimum and maximum values of R. G and B, one proceeds with the determination of the colours inside these ranges that generate *max_hue*; in fact, $max\_hue \pm delta$. These colours are stored on a table, called *colour table*. The *delta value* increases the range of acceptable hues by the system. It is defined as above, based on the entropy of the hues of the pixels (*h_hue*).

Selected the colours to be used in the final image, a new palette is created, called *the hue-scale palette*. For each entry of the colour table, one evaluates its corresponding grey value. As the difference between adjacent colours in the table is very small, there are few grey values. The grey palette is then replaced by another in which each grey value is exchanged by an entry of the colour table that may generate this grey value.

This technique modifies only a small part of the greyscale palette. To create a palette of similar colours, the other grey values are also defined based on the maximum value of the colour table that was inserted onto the grey palette.

Let *pos_min* and *pos_max* be the minimum and maximum position of the grey palette that was exchanged by entries of the colour table. Two counters are created: *count1* and *count2* initialized as zero and *pos_min*, respectively.

The entries on the grey palette with indexes less than *pos_min* have its colour exchanged by the last colour of the generated part of the palette minus *count2* which is then decremented of one.

Colours with indexes greater than *pos_max* are mapped onto the colour in the position pos_max added to count1, which is incremented by one.

In fact, as only the values of the grey palette are altered the structure of the image itself is not changed, increasing the performance of the technique.

As the values of the standard deviation are very small for the textual images, they are added to 5, defining an offset.

With this algorithm, the greyscale synthetic textual image can be colourised and added to the texture generating the final synthetic image. Figure 11 and 12 present examples of the application of the complete algorithm.

5 Conclusions

This paper presents a system for complete generation of a synthetic version of images of historical documents. An algorithm for automatic creation of texture is introduced and the settings for the building of the textual image from a text file produced by an OCR tool. The final synthetic image may be considered good both by qualitative and quantitative measures. The system was applied to a set of 30 document files and the synthetic images were analysed thru visual inspection and ANOVA. The PSNR was not analysed because the new image has different positions of the characters and the peak-to-noise ratio makes an analysis pixel-to-pixel.

The ANOVA method was applied to analyse the histogram of all three RGB components observing the mean, standard deviation, skewness and kurtosis.

In terms of compression, the results are very efficient as the original images stored in JPEG file format with 1% loss occupies 380 Kbytes in average and the text and binary data files occupies only less than 2 Kbytes.

References

[1] N.Abramson. *Information Theory and Coding*. McGraw-Hill Book, 1963.

[2] D.H.Ballard e C.M. Brown. *Computer Vision*. Prentice-Hall, 1982.

[3] R.Gonzalez e P. Wintz. *Digital Image Processing*. Addison Wesley, 1995.

[4] J. N. Kapur, *Measures of Information and their Applications*, John Wiley and Sons, 1994.

[5] R.D.Lins, M.S. Guimarães Neto, L.R. França Neto and L.G. Rosa. *An Environment for Processing Images of Historical Documents*. Microprocessing & Microprogramming, pp. 111-121, North-Holland, January, 1995.

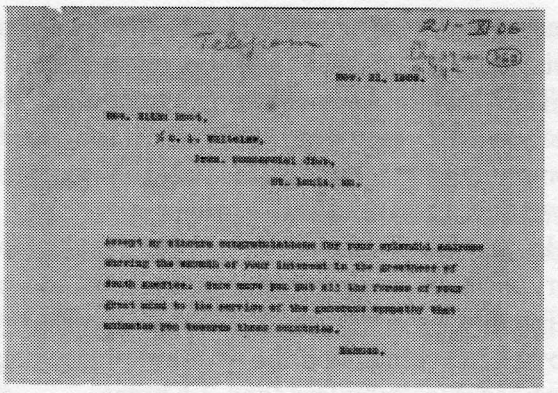
[6] C.A.B.Mello and R.D. Lins. *A New Segmentation Algorithm for True Colour Images of Historical Documents* (in portuguese), XVIII Simpósio Brasileiro de Telecomunicações, September, 2000, Brazil.

[7] C.A.B.Mello and R.D.Lins. *Image Segmentation of Historical Documents*, Visual 2000, August, 2000, Mexico.

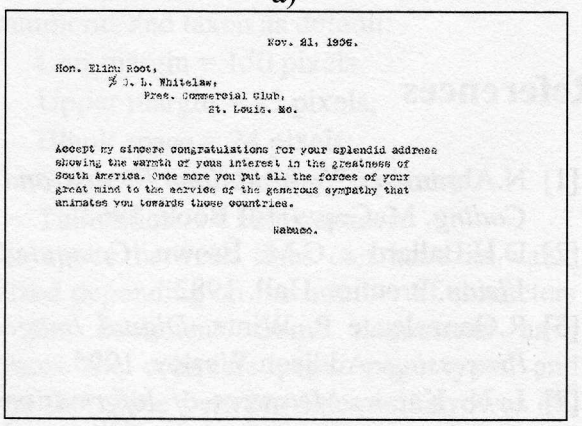
[8] C.A.B.Mello and R.D.Lins. *A Comparative Study on Commercial OCR Tools*.Proc. of Vision Interface'99, Canada, May, 1999.

[9] K.Sayood. *Introduction to Data Compression*. Morgan Kauffman, 1996.

[10] Nabuco Project: www.cin.ufpe.br/~nabuco



a)



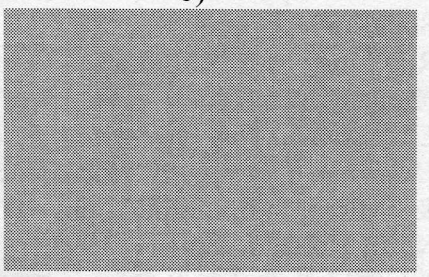
Nov. 21, 1906.

Hon. Elihu Root,
Whitelaw,
Pres. Commercial Club,
St. Louis. Mo.

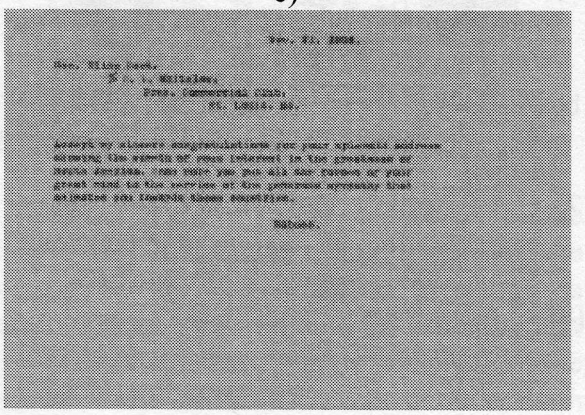
Accept my sincere congratulations for your splendid address showing the warmth of yous interest in the greatness of South America. Once more you put all the forces of your great mind to the service of the generous sympathy that animates you towards those countries.

Nabuco.

b)



c)



Nov. 21, 1906.

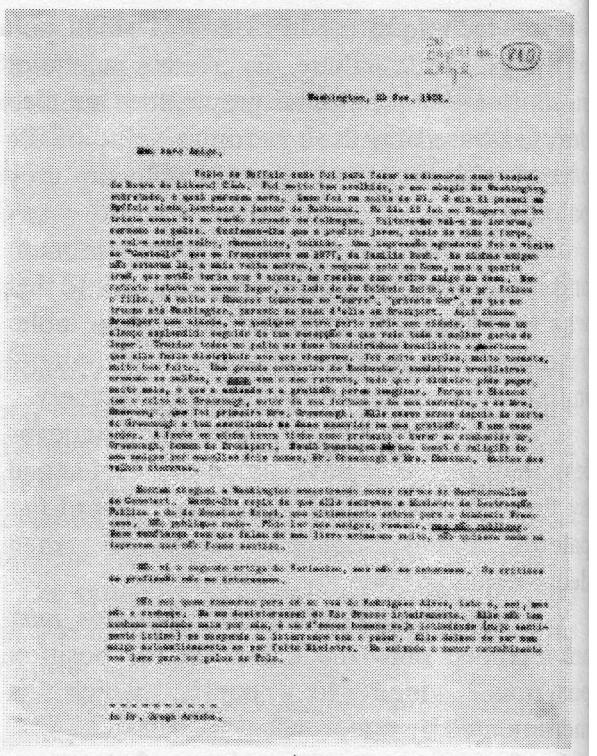
Hon. Elihu Root,
Whitelaw,
Pres. Commercial Club,
St. Louis. Mo.

Accept my sincere congratulations for your splendid address
showing the warmth of yous interest in the greatness of
South America. Once more you put all the forces of your
great mind to the service of the generous sympathy that
animates you towards those countries.

Nabuco.

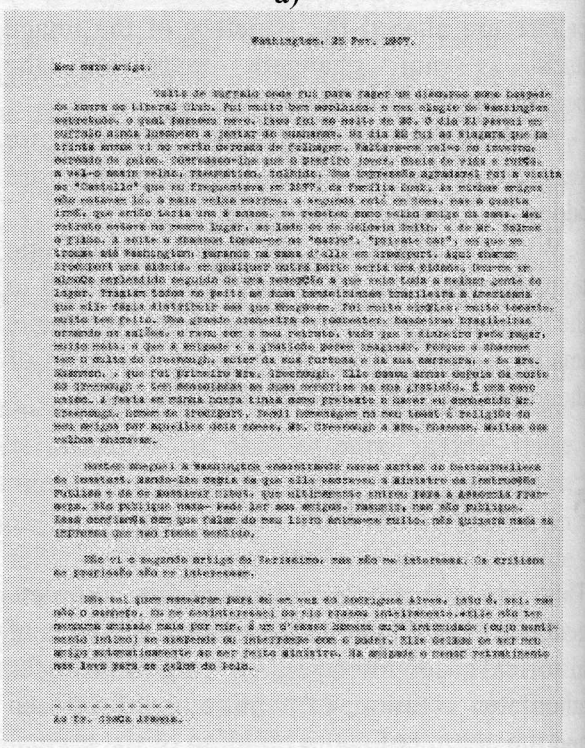
d)

Figure 11. Application of the generation scheme on the image of the document d0765. a) Original image, b) Textual image created after synthetisation and colourisation processes, c) Created texture and d) Final synthetic image.



Washington, 23 Fev. 1907.

a)



Washington, 23 Fev. 1907.

b)

Figure 12. Appliaction of the algorithm on the image of the document D0810-1. a) Original image and b) its synthetic version.