Automatic Borders Detection of Camera Document Images

N. Stamatopoulos, B. Gatos, A. Kesidis
Computational Intelligence Laboratory, Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos", GR-153 10 Athens, Greece
http://www.iit.demokritos.gr/cil/,
{nstam, bgat, akesidis}@iit.demokritos.gr

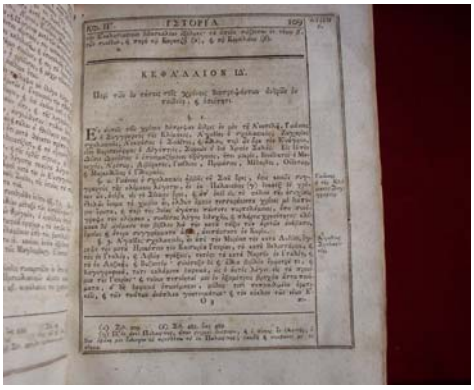
Abstract

When capturing a document using a digital camera, the resulting document image is often framed by a noisy black border or includes noisy text regions from neighbouring pages. In this paper, we present a novel technique for enhancing the document images captured by a digital camera by automatically detecting the document borders and cutting out noisy black borders as well as noisy text regions appearing from neighbouring pages. Our methodology is based on projection profiles combined with a connected component labelling process. Signal cross-correlation is also used in order to verify the detected noisy text areas. Experimental results on several camera document images, mainly historical, documents indicate the effectiveness of the proposed technique.

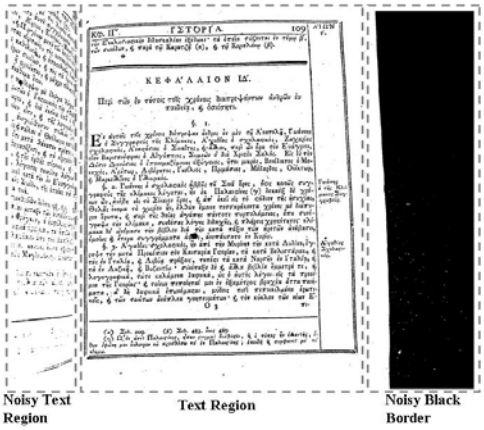
1. Introduction

Document images are often framed by a noisy black border or include noisy text regions from neighbouring pages when captured by a digital camera. Approaches proposed for document segmentation and character recognition usually consider ideal images without noise. However, there are many factors that may generate imperfect document images. When a page of a book is captured by a camera, text from an adjacent page may also be captured into the current page image. These unwanted regions are called "noisy text regions". Additionally, there will usually be black borders in the image. These unwanted regions are called "noisy black borders". Figure 1 shows an example of an image having noisy black borders as well as noisy text regions. All these problems influence the performance of segmentation and recognition processes.

There are only few techniques in the bibliography for page borders detection. Most of them detect only noisy black borders and not noisy text regions. Fan et al. [1] proposes a scheme to remove the black borders



(a)



(b)

Figure 1. Example of an image with noisy black border, noisy text region and text region. (a) Original camera document image (b) Binary document image

of scanned documents by reducing the resolution of the document image and by marginal noise detection and removal. Le and Thoma [2] propose a method for border removal which is based on classification of blank/textual/non-textual rows and columns, location of border objects, and an analysis of projection profiles

and crossing counts of textual squares. Avila and Lins [3] propose the invading and non-invading border algorithms which work as "flood–fill" algorithms. The invading algorithm, in contrast with non-invading, assumes that the noisy black border does not invade the black areas of the document. Finally, Avila and Lins [4] propose an algorithm based on "flood-fill" component labelling and region adjacency graphs for removing noisy black borders.

In this paper, a new and efficient algorithm for detecting and removing noisy black borders as well as noisy text regions is presented. This algorithm uses projection profiles and a connected component labelling process to detect page borders. Additionally, signal cross-correlation is used in order to verify the detected noisy text areas. The experimental results on several camera document images, mainly historical, documents indicate the effectiveness of the proposed technique. The rest of this paper is organized as follows. In section 2 the proposed technique is presented while experimental results are discussed in Section 3. Finally, conclusions are drawn in Section 4.

2. Proposed method

Before the noisy borders detection and removal takes place, we first proceed to image binarization using the efficient technique proposed in [5]. This technique does not require any parameter tuning by the user and can deal with degradations which occur due to shadows, non-uniform illumination, low contrast, large signal-dependent noise, smear and strain.

2.1. Noisy black border detection and removal

In this stage we detect and remove noisy black borders (vertical and horizontal) of the image. The proposed algorithm which is mainly based on horizontal and vertical profiles is described in the flowchart of Fig. 2. Our aim is to calculate the limits, XB1, XB2, YB1 and YB2, of text regions as shown in Fig. 3. In order to achieve this, we first proceed to an image smoothing, then calculate the starting and ending offsets of borders and text regions and then calculate the borders limits. The final clean image without the noisy black borders is calculated by using the connected components of the image.

Consider a binary image:

$$I(x,y)=\{0,1\} \quad 0 \le x < I_x, 0 \le y < I_y \qquad (1)$$

The main modules of the proposed technique are as follows.

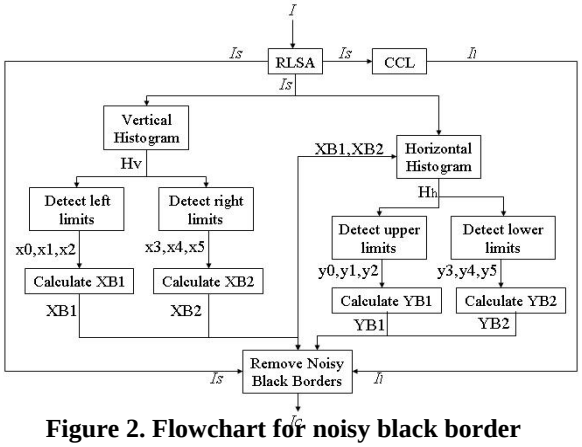


Figure 2. Flowchart for noisy black border detection and removal.

RLSA: Horizontal and vertical smoothing with the use of the Run Length Smoothing Algorithm (RLSA) [6]. RLSA examines the white runs existing in the horizontal and vertical direction. For each direction, white runs with length less than a threshold are eliminated. The empirical value of horizontal and vertical length threshold is 4 pixels. The resulting image is $I_s(x,y)$.

CCL (Connected Component Labeling): Calculate the connected components of the image $I_s(x,y)$ based on the approach described in [7]. The image consists of CS connected components C_i and the resulting labeled image is given by I_l:

$$I_l(x,y)=\begin{cases} i & \text{if } (x,y)\in C_i,\ 0<i<CS \\ 0 & \text{othewise} \end{cases} \qquad (2)$$

Figure 3. Limits XB1, XB2, YB1 and YB2 of text regions after noisy black border detection.

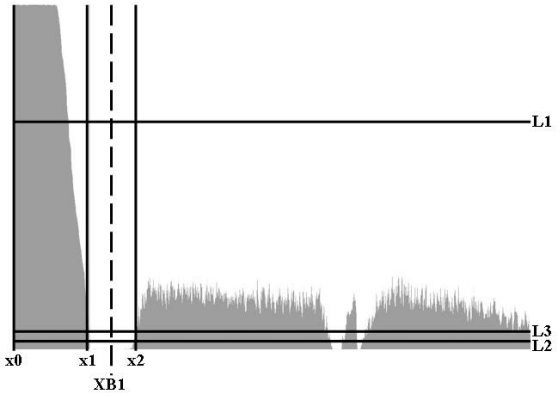


Figure 4. Projections of image in Fig. 3 and left limits detection.

Vertical Histogram: Calculate vertical histogram H_v, which is the sum of black pixels in each column.

$$H_v(x) = \sum_{y=0}^{Iy-1} I_s(x,y) \text{ where } 0 \le x < I_x \qquad (3)$$

Detect left limits: Detect vertical noisy black borders in the left side of the image (see Fig. 4).

Initially we search for the start and the end (x_0, x_1) of the left vertical black border. Calculate x_0 as follows:

$$x_0 = \min(x) : (H_v(x) > L_1) \; or \; (H_v(x) < L_2) \quad \text{where } 0 \le x < I_x / 5 \qquad (4)$$

The first condition, ($H_v(x) > L_1$), is satisfied when the black border starts from the left side of the image, which is the most usual case (see Fig. 3), while the second condition, ($H_v(x) < L_2$), is satisfied when white region exists before the black border. If we don't find any x_0 that satisfies the above conditions we set $x_0 = -1$, $x_1 = -1$, $x_2 = -1$ and stop this process. Otherwise, x_1 is calculated as follows:

$$x_1 = \begin{cases} \min(x) : H_v(x) < L_2 \;,\; x_0 < x < I_x/2 \text{ if } H_v(x_0) > L_1 \\ \min(x) : H_v(x) > L_1 \;,\; x_0 < x < I_x/2 \text{ othewise} \end{cases} \qquad (5)$$

If we don't find any x_1 that satisfies the conditions we set $x_0 = -1$, $x_1 = -1$, $x_2 = -1$ and stop this process.

After we have located the black border we search for the start (x_2) of the text region (see Fig. 4) and calculate it as follows:

$$x_2 = \min(x) : \; H_v(x) < L_1 \text{ AND } H_v(x) > L_3, \; x_1 < x < I_x/2 \qquad (6)$$

If these is no x_2 satisfying Eq. (6) we set $x_2 = -1$. After experimentations, the values of L_1, L_2 and L_3 are set to: $L_1 = (2/3) * I_y$, $L_2 = (1/50) * I_y$, $L_3 = (1/20) * I_y$.

Calculate XB1: Calculate left limit (XB1) of text regions (see Fig. 3) as follows:

$$XB1 = \begin{cases} 0 & \text{if } x_0 = -1 \\ x_0 + (x_1 - x_0)/2 & \text{if } x_2 = -1 \\ x_1 + (x_2 - x_1)/2 & \text{if } x_2 \ne -1 \end{cases} \qquad (7)$$

A similar process is applied in order to detect the vertical noisy black border of the right side of the image as well as the right limit XB2 of text regions.

Horizontal Histogram: Calculate horizontal histogram H_h, which is the sum of black pixels in each row at XB1 and XB2 limits.

$$H_h(y) = \sum_{x=XB1}^{XB2} I_s(x,y) \text{ where } 0 \le y < I_y \qquad (8)$$

A similar process as for the vertical noisy black borders is applied in order to detect the horizontal noisy black borders as well as the upper (YB1) and bottom (YB2) limits of text regions (see Fig. 3).

Remove Noisy Black Borders: All black pixels that belong in a connected component C_i which includes at least one pixel that is out of limits are transformed in white. Finally, we get the image $I_C(x,y)$ as follows:

$$I_C(x,y) = \begin{cases} 0 & \text{if } I_1(x,y)=i \text{ and } \exists\, (x1,y1) : (x1 \le XB1 \text{ or} \\ & x1 \ge XB2 \text{ or } y1 \le YB1 \text{ or } y1 \ge YB2) \text{ and} \\ & I_1(x1,y1)=i \\ I(x,y) & \text{otherwise} \end{cases} \qquad (9)$$

2.2. Noisy text region detection and removal

In this stage, we detect noisy text regions of the image $I_c(x,y)$ that resulted from the previous stage. The flowchart of our algorithm is shown in Fig. 5. Before it, we proceed to skew correction based on [8]. Our aim is to calculate the limits, *XT1* and *XT2*, of the text region as shown in Fig. 6. We first proceed to an

image smoothing in order to connect all pixels that belong to the same line. Then, we calculate the vertical histogram in order to detect text zones. Finally, we detect noisy text regions with the help of the signal cross-correlation function. The main modules of the proposed technique are described in detail as follows.

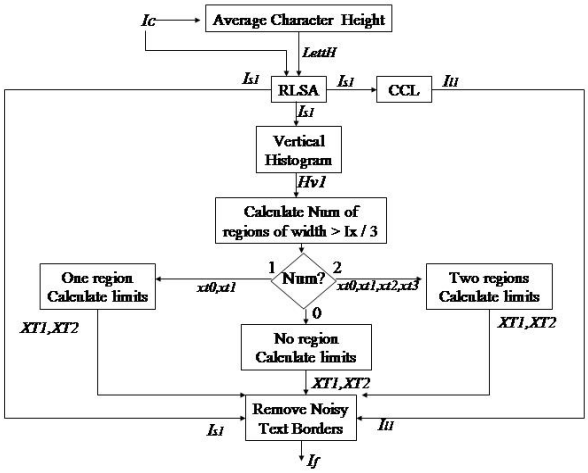


Figure 5. Flowchart for noisy text region detection and removal.

Average Character Height: The average character height (*LettH)* for the document image is calculated based on [9].

RLSA: Horizontal and vertical smoothing with the use of the RLSA [6] by using dynamic parameters which depend on average character height (LettH). Our aim is to connect all pixels that belong to the same line. The horizontal length threshold is experimentally defined as the LettH while the vertical length threshold is experimentally defined as 50% of the LettH. The resulting image is $I_{s1}(x,y)$.

CCL (Connected Component Labeling): Extract the connected components of the image [7]. The image consists of CS connected components C_i and the resulting image is $I_{l1}(x,y)$ as in Eq. (2).

Vertical Histogram: Calculate vertical histogram H_{v1} as follows:

$$H_{v1}(x) = \sum_{y} I_{s1}(x,y) \qquad (10)$$

Calculate Number of regions of width $> I_x^{'}/3$ **:** Check if the number of consecutive *x*, where $H_{v1}(x) > L_4$, is greater than $W = (1/3) * I_x^{'}$ (where $I_x^{'} = XB2 - XB1$) and calculate the number of regions that satisfy this condition. Let suppose that two regions have been found and let xt_0, xt_1 and xt_2, xt_3 denote the regions' limits, as shown in Fig. 7. Similarly, if one region has been found we set xt_0, xt_1 to denote the region's limits.

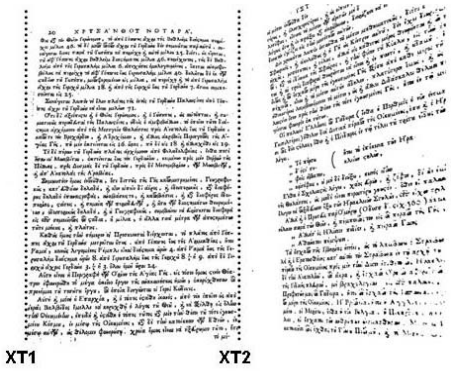


Figure 6. Limits XT1 and XT2 of text region after noisy text region detection.

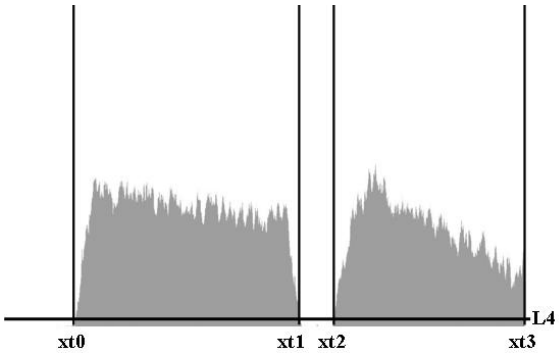


Figure 7. Projections of image in Fig. 5 and text regions detection.

Two regions-Calculate Limits: We examine if one of these regions is a noisy text region. Calculate signal cross-correlation for each region (SC_0 , SC_1) [10]. First, we calculate SC_y (Eq. 11) for each line of the region

$$SC(a,y) = 1 - \frac{2}{M}\sum_{k=0}^{M}(I_{s1}(k,y) \text{ XOR } I_{s1}(k,y+a)) \qquad (11)$$

where M is the region's width and *a* is the distance between two lines. Finally, total SC_i of region *i*, is the middle count of all SC_y .

Then, we calculate limits *XT1* and *XT2* as follows:

$$\begin{aligned}&\text{if } (SC_0 < 0.5 \text{ AND } SC_1 < 0.5) \text{ then}\\&\quad (XT1 = xt0 \text{ AND } XT2 = xt3)\\&\text{else if } (SC_0 < SC_1) \text{ then}\\&\quad (XT1 = xt0 \text{ AND } XT2 = xt1)\\&\text{else}\\&\quad (XT1 = xt2 \text{ AND } XT2 = xt3)\end{aligned} \tag{12}$$

One region-Calculate Limits: We examine if the noisy text region and the text region are very close to each other without leaving a blank line between them. If the width of region is less than 70% of $I_x^{'}$ we consider that we don't have noisy text region, so $XT1 = xt_0$ *and* $XT2 = xt_1$. Otherwise, we divide it into eight regions and calculate the signal cross-correlation for each region ($SC_1,...,SC_8$) using Eq. 11.

Calculate XT1 and XT2 as follows:

- If $SC_1 < 0.5$ and $SC_8 < 0.5$ we don't have noisy text region, so $XT1 = xt_0$ and $XT2 = xt_1$.
- If $SC_1 > 0.5$ we search for the last consecutive region *i* where $SC_i > 0.5$ and we find an x' where H_{v1} is minimum in this region.
  If $(xt_1 - x') \geq W$ then
  $XT1 = x'$ and $XT2 = xt_1$
  else
  $XT1 = xt_0$ and $XT2 = xt_1$
- If $SC_8 > 0.5$ we search for the last consecutive region *i* where $SC_i > 0.5$ and we find an x' where H_{v1} is minimum in this region.
  If $(x' - xt_0) \geq W$ then
  $XT1 = xt_0$ and $XT2 = x'$
  else
  $XT1 = xt_0$ and $XT2 = xt_1$

No region-Calculate Limits: In this case, the text region consists of two or more columns and we try to locate and separate them from the noise text regions, if these exist. First, we check if the number of consecutive x, where $H_{v1}(x) > L_4$, is greater than $W/4$. If two or less regions are find that satisfy the conditions, we set *XT1=XB1* and *XT2=XB2*. If we find three or more regions that satisfy the conditions we calculate the signal cross-correlation (Eq. 11) for the left and the right region (SC_0 , SC_1). Consider that left region's limits are xt_0 and xt_1 and the right region's limits are xt_2 and xt_3. We calculate *XT1* and *XT2* as in Eq. 12.

Remove Noisy Text Region: All black pixels that belong in a connected component C_i which does not include at least one pixel in the limits *XT1*and *XT2* are transformed in white. The final image $I_f(x,y)$ is calculated as follows:

$$I_f(x,y) = \begin{cases} I_c(x,y) & I_{l1}(x,y) = i \text{ AND } \exists\,(x1,y1): \\ & (x1 \geq XT1 \text{ or } x1 \leq XT2) \text{ AND } I_{l1}(x1,y1) = i \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

3. Experimental results

To verify the validity of the proposed method, experiments were conducted on several camera document images. We used 1705 document images mainly consisting of historical documents that contain noisy black borders as well as noisy text region appearing from adjacent pages. After visual checking, we found that in 1344 images (78,82% of testing set) the noisy black borders and the noisy text region were correctly removed. Fig. 8 depicts some examples of document images illustrating the page borders detection and removal processes. Difficulties arise in two cases. First, the text region and the noisy text region may be very close to each other without any blank line between them. In this case, a part or even a whole noisy text region may still remain in the resulting image. The second, and perhaps even more difficult case, is when the noisy black border merges with the text region. This may lead to loss of information.

In order to compare our approach with other state-of-the-art approaches, we implemented the methods of Fan et al. [1], Avila and Lins [3] (invading and non-invading algorithms) and used the implementation of the recent algorithm of Avila and Lins [4] found in [11] (see Figs. 9,10). All these methods have been proposed to remove only noisy black borders and not noisy text regions. In the first example we see that only Fan's method and the invading algorithm can effectively remove the noisy black border. Moreover, in the second example, in which noisy black borders are not continuous, none of these methods can effectively remove it.

4. Conclusion

This paper proposes an effective algorithm for the detection and removal of noisy black borders and noisy text regions inserted on document images that are captured by a digital camera. The new algorithm is based on projection profiles combined with a

connected component labelling process. Additionally, signal cross-correlation is used in order to verify the detected noisy text areas. Experimentations in several camera document images prove the effectiveness of the proposed methodology. Our future research will focus on the optimization of the proposed method in the cases, when the noisy black border merges with the text region and when the text region and the noisy text region are very close to each other, as described in section 3.

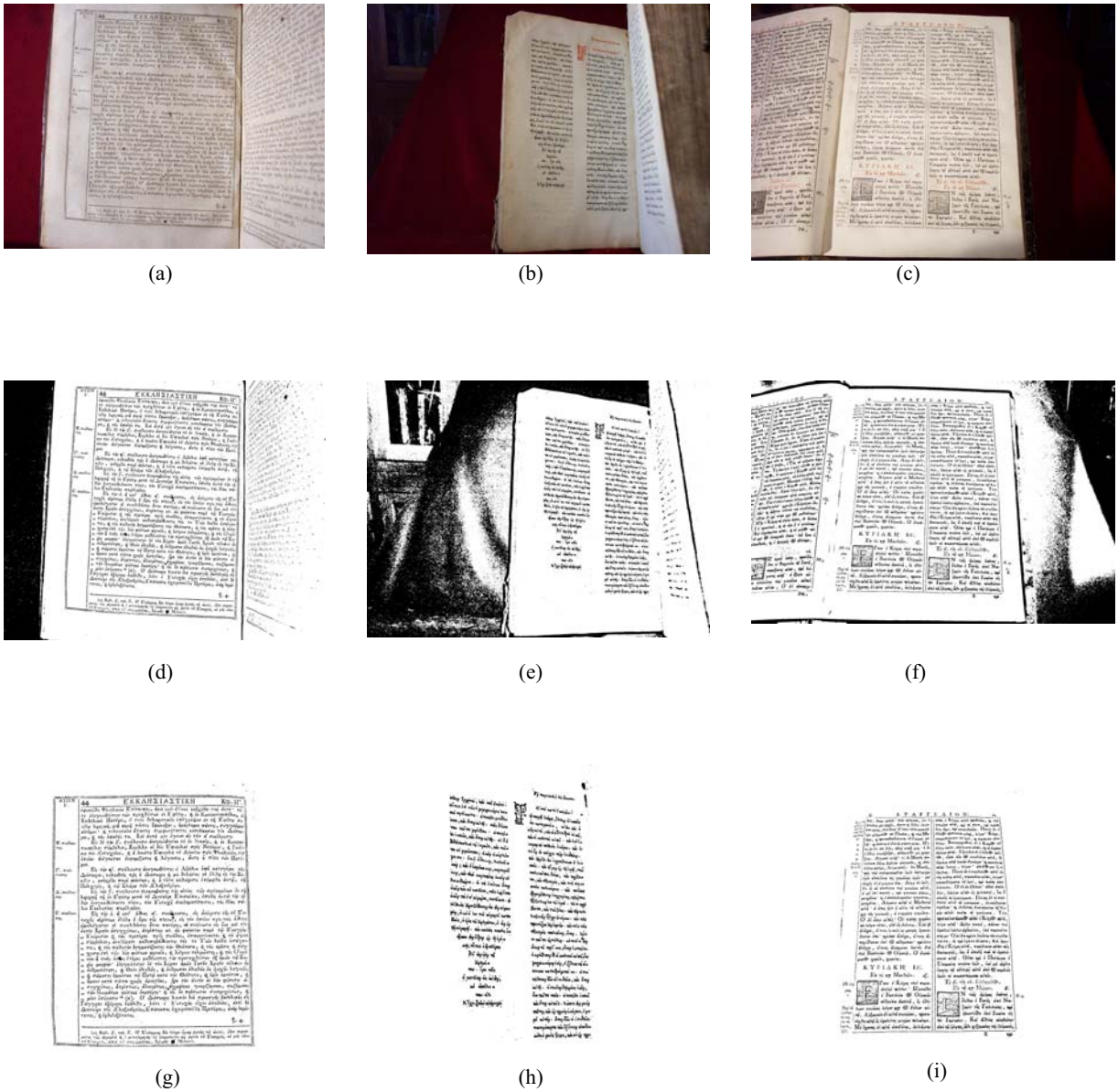
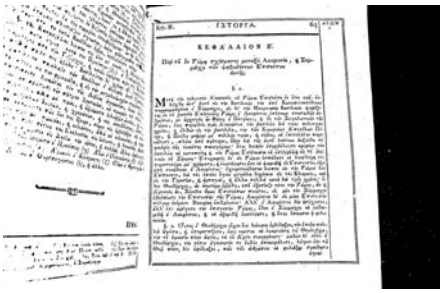
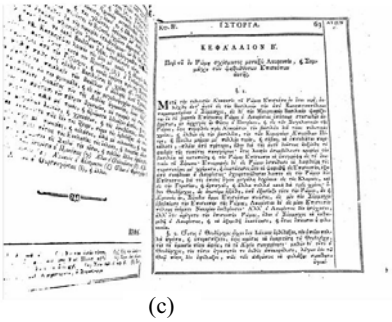


Figure 8. (a)-(b)-(c) Original camera document images, (d)-(e)-(f) binary images, (g)-(h)-(i) results of proposed method.



(a)

(b)



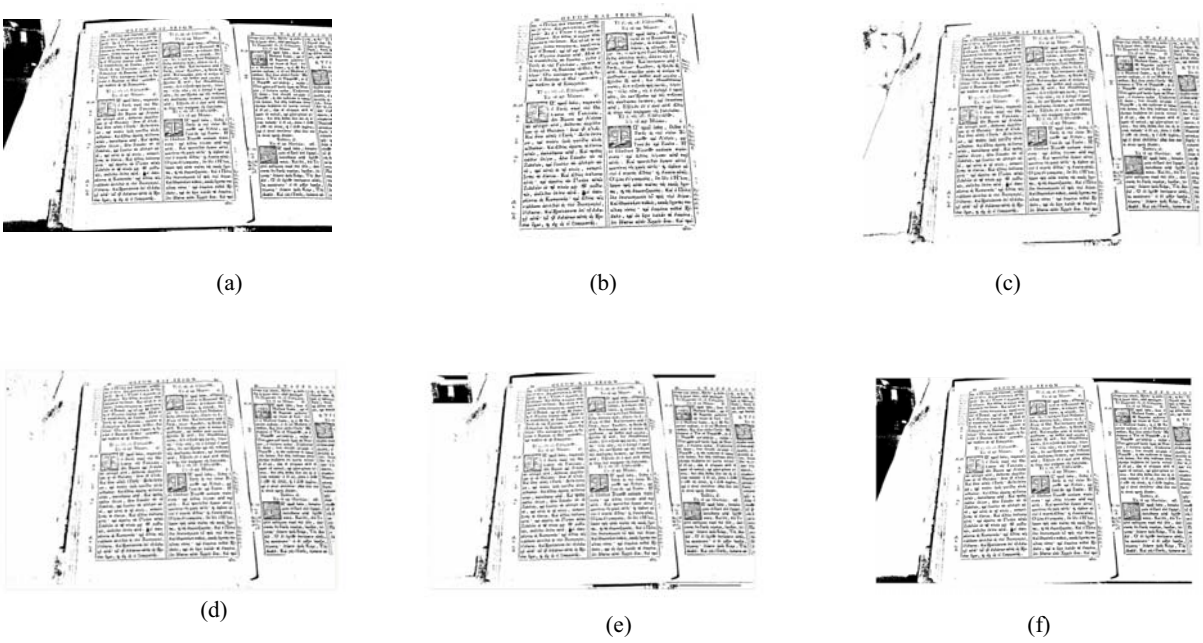
(c)

(d)

(e)

(f)

Figure 9: (a) Original image (b) proposed method (c) method of Fan et al. (d) invading algorithm (e) non-invading algorithm (f) an approach of the algorithm of Avila and Lins.



(a) (b) (c)

(d) (e) (f)

Figure 10: (a) Original image (b) proposed method (c) method of Fan et al. (d) invading algorithm (e) non-invading algorithm (f) an approach of the algorithm of Avila and Lins.

5. References

[1] Kuo-Chin Fan, Yuan-Kai Wang, Tsann-Ran Lay, "Marginal Noise Removal of Document Images", Pattern Recognition, 35(11), 2002, pp. 2593-2611.

[2] D.X. Le, G.R. Thoma, "Automated Borders Detection and Adaptive Segmentation for Binary Document Images", International Conference on Pattern Recognition, 1996, p. III: 737-741.

[3] B.T. Avila and R.D. Lins, "A New Algorithm for Removing Noisy Borders from Monochromatic Documents", Proc. of ACM-SAC'2004, Cyprus, ACM Press, March 2004, pp 1219-1225.

[4] B.*T. Avila, R.D. Lins, "*Efficient Removal of Noisy Borders from Monochromatic Documents", ICIAR 2004, LNCS 3212, 2004, pp. 249-256.

[5] B. Gatos, I. Pratikakis and S. J. Perantonis, "Adaptive Degraded Document Image Binarization", *Pattern Recognition*, Vol. 39, 2006, pp. 317-327.

[6] Wahl, F.M., Wong, K.Y., and Casey R.G.: "Block Segmentation and Text Extraction in Mixed Text/Image Documents", Computer Graphics and Image Processing, 20, 1982, pp 375-390.

[7] Fu Chang, Chun-Jen Chen, Chi-Jen Lu, "A linear-time component-labeling algorithm using contour tracing technique", *Computer Vision and Image Understanding, Vol. 93, No.2,* February 2004, pp. 206-220.

[8] B. Gatos, N. Papamarkos and C. Chamzas, "Skew detection and text line position determination in digitized documents", *Pattern Recognition*, Vol. 30, No. 9, 1997, pp. 1505-1519.

[9] B. Gatos, T. Konidaris, K. Ntzios, I. Pratikakis and S. J. Perantonis, "A Segmentation-free Approach for Keyword Search in Historical Typewritten Documents", *ICDAR*, Seoul, Korea, August 2005, pp. 54-58.

[10] Sauvola J., Pietikainen, M.: "Page segmentation and classification using fast feature extraction and connectivity analysis", ICDAR, 1995, pp. 1127-1131.

[11] Software of Personal PC Helpers (http://www. sharewareconnection.com/bordershelper.htm).