Automatic Table Detection in Document Images

B. Gatos, D. Danatsas, I. Pratikakis and S. J. Perantonis

Computational Intelligence Laboratory, Institute of Informatics and Telecommunications
National Center for Scientific Research "Demokritos"
GR 15310 Athens, Greece
{bgat, dan, ipratika, sper}@iit.demokritos.gr

Abstract. In this paper, we propose a novel technique for automatic table detection in document images. Lines and tables are among the most frequent graphic, non-textual entities in documents and their detection is directly related to the OCR performance as well as to the document layout description. We propose a workflow for table detection that comprises three distinct steps: (i) image pre-processing; (ii) horizontal and vertical line detection and (iii) table detection. The efficiency of the proposed method is demonstrated by using a performance evaluation scheme which considers a great variety of documents such as forms, newspapers/magazines, scientific journals, tickets/bank cheques, certificates and handwritten documents.

1 Introduction

Nowadays, we experience a proliferation of documents which leads to an increasing demand for automation in document image analysis and processing. Automatic detection of subsequent page components like tables gives a great support to fulfill the demand for automation. More specifically, in the case of a table recovery, a great support to compression, editing and information retrieval purposes can be given.

Tables have physical and logical structure [1]. The physical structure concerns the location in an image of all the constituent parts of a table. The logical structure defines the type of the constituent parts and how they form a table. Therefore, all parts in a table have both physical and logical structure.

In this paper, we focus on the detection of all lines, both vertical and horizontal, along with their intersection, which will aid not only to detect a table which consequently can be extracted out of a whole document but also to describe both the physical and logical structure, thus, inferring a table recognition process.

In the literature, other researches have worked to accomplish the goals mentioned above. Zheng et al. [2] proposed a frame line detection algorithm based on the Directional Single-Connected Chain (DSCC). Each extracted DSCC represents a line segment and multiple non-overlapped DSCCs are merged to compose a line based on rules. During our experiments, we have compared this approach with our proposed approach for horizontal/vertical table line detection. Neves and Facon [3] have presented a method for automatic extraction of the contents of passive and/or active cells

in forms. This approach is based on the analysis and recognition of the types of intersection of the lines that make up the cells. In the particular domain of business letters, Kieninger and Dengel [4] propose the so-called T-Recs Table location that consists of block segmentation and table locator. The table locator is based on simple heuristics that concern the extracted blocks. Finally, Cesarini et al. [5] describe an approach for table location in document images where the presence of a table is hypothesized by searching parallel lines in the modified X-Y tree of the page. Furthermore, located tables can be merged on the basis of proximity and similarity criteria.

In this paper, we propose a novel technique for automatic table detection in document images that neither requires any training phase nor uses domain-specific heuristics, thus, resulting to an approach applied to a variety of document types. Experimental results support the robustness of the method. The proposed approach builds upon several consequent stages that can be mainly identified to the following: (i) image preprocessing; (ii) horizontal and vertical line detection and (iii) table detection. In the following sections, we present our methodology for table detection in document images, as well as our experimental results that demonstrate the efficiency of the proposed method.

2 Methodology

2.1 Pre-processing

Pre-processing of the document image is essential before proceeding to the line and table detection stages. It mainly involves image binarization and enhancement, orientation and skew correction as well as noisy border removal. Binarization is the starting step of most document image analysis systems and refers to the conversion of the gray-scale image to a binary image. The proposed scheme for image binarization and enhancement is described in [6]. It is an adaptive approach suitable for documents with degradations which occur due to shadows, non-uniform illumination, low contrast, large signal-dependent noise, smear and strain. Text orientation is determined by applying an horizontal/vertical smoothing, followed by a calculation procedure of vertical/horizontal black and white transitions [7]. The proposed scheme for skew correction is described in [8] and uses a fast Hough transform approach based on the description of binary images using rectangular blocks. In the pre-processing stage of our approach, the process of noisy borders removal is based on [9] and employs a "flood-fill" based algorithm that starts expanding from the outside noisy surrounding border towards the text region. Fig. 1 illustrates the proposed pre-processing step.

In the proposed methodology, we use a particular parameterization that depends on the average character height of the document image. Therefore, we proceed with an average character size estimation step that is more specifically required for adjusting all line detection algorithm parameters in order to achieve invariance to the scanning resolution or the character font size. Our main intentions are to exclude all short line segments that belong to character strokes and to approximate the maximum expected line thickness. We propose a method to automatically estimate the average character

height based on calculating the surrounding rectangles height of the image connected components. We take the following steps:

STEP 1: We pick a random pixel (x,y) that has at least one background pixel in its 4 connected neighborhood.

STEP 2: Starting from pixel (x,y), we follow the contour of the connected component that pixel (x,y) belongs to.

STEP 3: We repeat steps 1, 2 for all existing connected components until we have a maximum number of samples (*MaxSamples*). During this process we calculate the histogram H_h of the surrounding rectangles height h at the corresponding connected components.

STEP 4: We compute the maximum value of the histogram H_h which expresses the average character height *AH*. An example of the estimated average character size is illustrated in Figure 2.
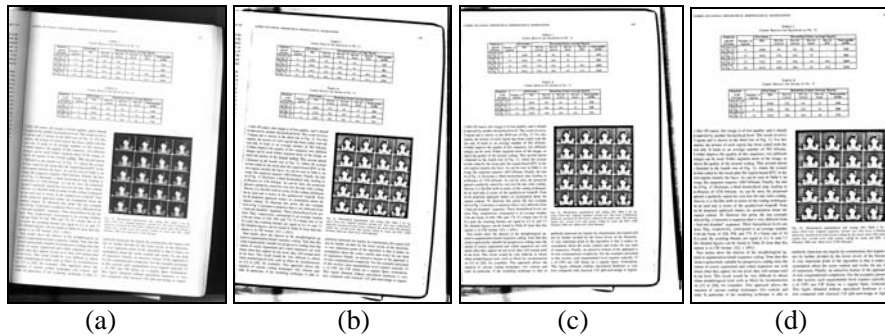

Fig. 1. Document image pre-processing example. (a) Original gray scale image; (b) resulting image after binarization and image enhancement; (c) resulting image after skew correction; (d) resulting image after noisy border removal.
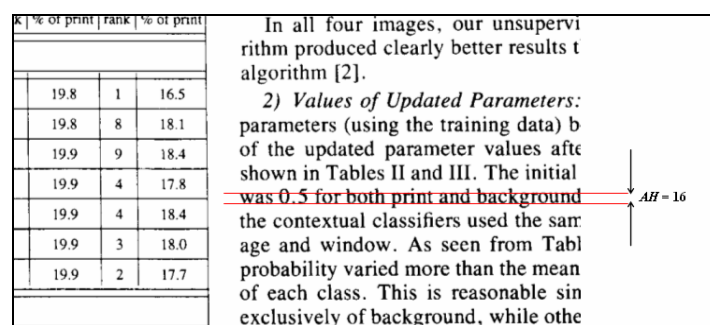

Fig. 2. The estimated average character height of a document image.

2.2 Line Detection

A novel technique for horizontal and vertical line detection in document images is proposed. The technique is mainly based on horizontal and vertical black runs processing as well as on image/text areas estimation in order to exclude line segments that belong to these areas. Initially, a set of morphological operations with suitable structuring elements is performed in order to connect possible line breaks and to enhance line segments. The distinct steps of the proposed line detection technique are the following: (i) horizontal and vertical line estimation and (ii) line estimation improvement by using image/text areas removal.

Horizontal and vertical lines estimation. At this step, we make a first estimation of horizontal and vertical line segments. The final estimation of lines will be accomplished after a refinement of this result by removing line segments that belong to image/text areas. The proposed line detection algorithm is based on horizontal and vertical black runs processing as well as on a set of morphological operations with suitable structuring elements in order to connect possible line breaks and to enhance line segments. All parameters used in this step depend on the average character height AH that has been calculated in Section 2.1. Starting with the binary image IM (with 1s that corresponds to text regions and 0s to background regions), we take the following steps:

STEP 1: We proceed to a set of morphological operations of the image IM with suitable structuring elements. Our intention is to connect line breaks or dotted lines but not to connect neighboring characters (see Fig. 3). We calculate images IM_H and IM_V for horizontal and vertical line detection, respectively, as in the following Eq. 1, 2:

$$IM_H = IM \cup (((IM \ominus B_{HR}) \cup (IM \ominus B_{HL})) \oplus B_H),$$

(1)

$$\text{where } B_{HR} = \overset{\xleftarrow{\quad AH \quad}}{[111\ldots \boxed{1}]}, B_{HL} = \overset{\xleftarrow{\quad AH \quad}}{[\boxed{1}\ldots 111]}, B_H = \begin{bmatrix} 1 & . & . & . & 1 \\ . & . & \boxed{1} & . & . \\ 1 & . & . & . & 1 \end{bmatrix} \overset{\xleftarrow{\;0.5AH+1\;}}{\Big\updownarrow} 0.2AH+1$$

$$IM_V = IM \cup (((IM \ominus B_{VD}) \cup (IM \ominus B_{VU})) \oplus B_V),$$

(2)

$$\text{where } B_{VD} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ . \\ . \\ . \\ \boxed{1} \end{bmatrix} \Big\updownarrow 1.2\,AH, \quad B_{VU} = \begin{bmatrix} \boxed{1} \\ . \\ . \\ . \\ 1 \\ 1 \\ 1 \end{bmatrix} \Big\updownarrow 1.2\,AH, \quad B_V = \begin{bmatrix} 1 & . & 1 \\ . & . & . \\ . & \boxed{1} & . \\ . & . & . \\ 1 & . & 1 \end{bmatrix} \overset{\xleftarrow{\,0.2AH+1\,}}{\Big\updownarrow} 0.5AH+1$$

	MAP1			MAP1			MAP1
μ_p	101.2		μ_p	101.2		μ_p	101.2
	(a)			(b)			(c)

Fig. 3. Morphological processing in order is to connect line breaks or dotted lines. (a) Initial image *IM*; (b) Resulting image IM_H; (c) Resulting image IM_V.

STEP2: All 1s of images IM_H and IM_V that belong to line segments of great length and small width are turned to a label values **L**. In the case of horizontal lines, all 1s of IM_H that belong to horizontal black runs of length greater than *AH* and to vertical black runs of length less than *AH* are turned to **L**. In the case of vertical lines, all 1s of IM_V that belong to vertical black runs of length greater than *AH* and to horizontal black runs of length less than *AH* are turned to **L**.

STEP3: Images IM_H and IM_V are smoothed in horizontal and vertical directions correspondingly in order to set to **L** all short runs that have a value different than **L**. In the case of horizontal lines, horizontal runs of IM_H pixels with values not equal to **L** and length less than *AH* are set to **L**. In the case of vertical lines, vertical runs of IM_V pixels with values not equal to **L** and length less than *AH* are set to **L**.

STEP 4: Horizontal and vertical lines in images IM_H and IM_V, respectively, are defined from all connected components with **L**-valued pixels having length greater than 2 *AH*.

Line estimation improvement by using Image/text areas removal. Image/text areas estimation is accomplished by performing an horizontal and vertical smoothing of image IM_n that has 1's for pixels that do not belong to the detected horizontal or vertical lines. After this smoothing, all connected components of great height ($> 3AH$) belong to graphics, images or text. In this phase, tables will not appear as individual connected components in the final smoothed image since vertical and horizontal lines are excluded. More specifically, we take the following steps:

STEP 1: We proceed to an horizontal smoothing of image IM_V by setting all horizontal runs with 0's that have length less than 1.2*AH* to **L**.

STEP 2: We proceed to a vertical smoothing by setting all vertical runs with 0's that have length less than 1.2*AH* to **L**.

STEP 3: Image/text areas *IT* are defined from **L**-valued connected components in the resulting IM_V image having surrounding rectangle height greater that 3 *AH*.

From all horizontal lines *HL* and vertical lines *VL* we exclude those that lie inside non line entities *IT* that have been estimated in the previous step. Fig. 4 illustrates the line detection step.
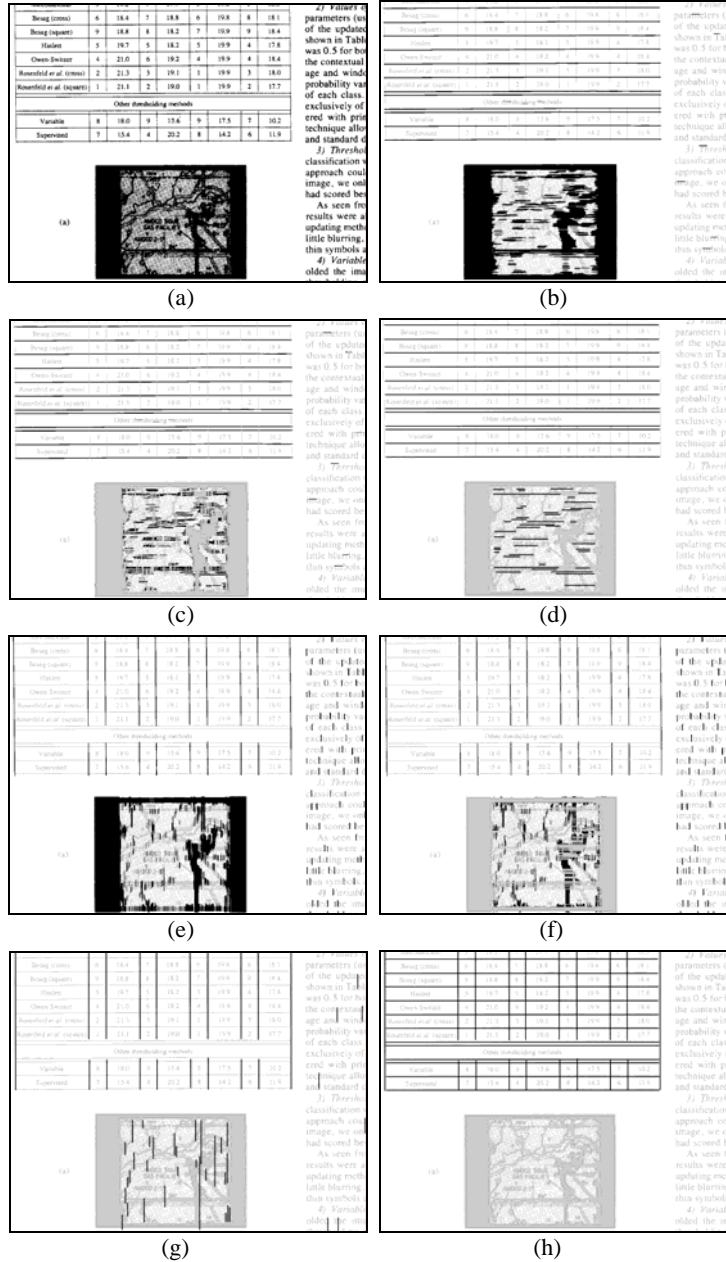
Fig. 4. Horizontal and vertical line estimation: (a) Initial image; (b,e) horizontal and vertical line segments of great length; (c,f) line segments of great length and small width; (d,g) detected horizontal and vertical lines after smoothing; (h) line estimation improvement using image/text areas removal.

2.3   Table Detection

After horizontal and vertical line detection we proceed to table detection. Our table detection technique involves two distinct steps: (i) Detection of line intersections and (ii) table detection - reconstruction.

Line estimation improvement by using Image/text areas removal. All possible line intersections (see Table 1) are detected progressively according to the following algorithm. First, we detect all intersections with IDs 1-4. In this case, an end point of an horizontal line and another end point of a vertical line define a line intersection of this type if they have the minimum distance among others around a neighborhood. Thereafter, we trace for intersections with IDs 5-8. In this case, an end point of either an horizontal line or a vertical line is tested against another line point which is not an end point and corresponds to a vertical line or an horizontal line, respectively. A line intersection of this type is defined for such points that have the minimum distance among others around a neighborhood. Finally, we detect intersections with ID 9 that correspond to horizontal and vertical line crossing points.

Table 1. Line intersections

ID	1	2	3	4	5	6	7	8	9
Line inter-sections									

Table detection and reconstruction. Table detection and reconstruction involves the following steps: First, all pixels that belong to the detected lines are removed (see Fig. 5(c)). Then, all detected line intersections are grouped first horizontally and then vertically. Each group is further aligned according to the mean value of the vertical or horizontal positions for horizontal and vertical groupings, respectively. Finally, we achieve a table reconstruction by drawing the corresponding horizontal and vertical lines that connect all line intersection pairs. Table detection and reconstruction is illustrated in Fig. 5.

3   Experimental Results

The corpus for the evaluation of the proposed methodology was prepared by selecting 102 images with a total of 2813 ground-truthed horizontal and vertical lines. It consists of scanned forms, newspaper - magazines, scientific papers, tickets – bank checks, certificates and handwritten documents. Most of the images have severe problems such as poor quality, broken lines or overlapping text and line areas. Representative results of the proposed methodology for line and table detection are illustrated in Fig. 6. In order to extract some quantitative results for the efficiency of the proposed methodology, we calculated the recognition rate and the recognition accuracy for horizontal and vertical line detection and compared the results with those of the DSCC algorithm [2] which is a state-of-the-art algorithm for unsupervised horizon-

tal/vertical table line detection and the corresponding source code is available at [11]. The performance evaluation method used is based on counting the number of matches between the detected horizontal/vertical lines and the corresponding horizontal/vertical lines appearing in the ground truth [10]. We use a "MatchScore" table for horizontal and vertical lines whose values are calculated according to the intersection of the resulting line pixels and the ground truth. A global performance metric can be detected if we combine the detection rate and the recognition accuracy results according to the following formula:

$$GlobalPerformanceMetric = \frac{2\,DetectionRate * \mathrm{Re}\,cognitionAccuracy}{DetectionRate + \mathrm{Re}\,cognitionAccuracy} \qquad (3)$$

sults of all the thinning algorithms.

TABLE II
STATISTICS ON THINNING ALGORITHMS
(FOR 1,000 CHARACTERS IN DATABASES 1 AND 2)

Algorithm	Database 1				Database 2			
	Time (sec)	Extra Pixels	End Points	Primi-tives	Time (sec)	Extra Pixels	End Points	Primi-tives
Chen & Hsu [3]	39.0	373	2109	2295	184	431	2284	2440
Chen & Tsai [5]	54.3	19072	2189	2523	179	36369	2541	2510
Chin et al. [6]	15.0	2342	2145	2238	65	3578	2494	2454
Ono & Hall [8]	9.5	1004	2186	2315	41	968	2351	2468
Hilditch [12]	12.9	0	2221	2337	60	0	2384	2486
Holt et al. [13]	12.2	7387	2119	2294	49	13828	2281	2442
Suzuki & Abe [23]	3.9	0	2250	2330	14	0	2447	2511
Wang & Zhang [25]	9.3	1111	2085	2287	41	1118	2229	2438
Wu & Tsai [26]	12.4	433	2190	2305	53	489	2360	2466
Zhang & Suen [28]	10.0	8538	2084	2286	42	15359	2232	2437

C. Number of End Points
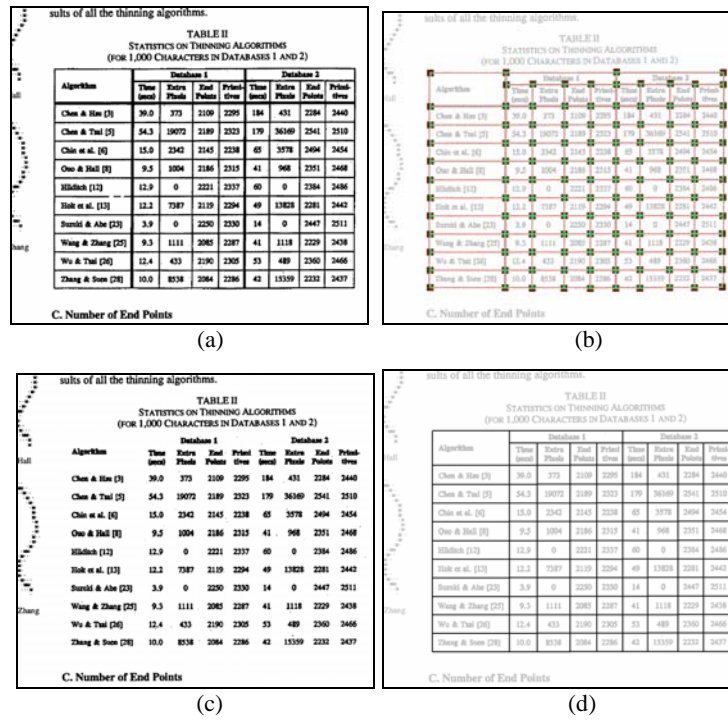
(a)   (b)   (c)   (d)

Fig. 5. Table detection and reconstruction: (a) Initial image; (b) detected line intersections; (c) image without horizontal and vertical lines; (d) table reconstruction.

As shown at Fig.7, for all types of the examined scanned documents, we get higher recognition rates compared to the DSCC algorithm. The global performance metric for all images is 80.6%, while DSCC algorithm achieves 70.1%.
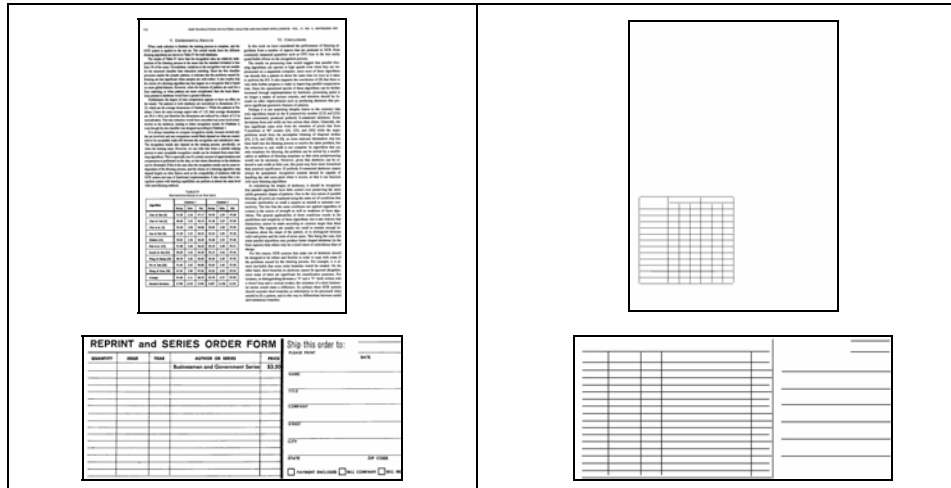
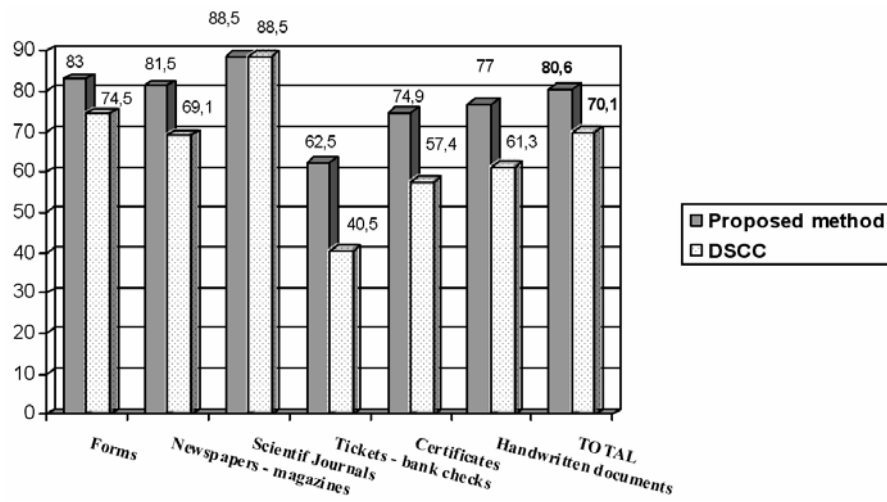**Fig. 6.** Line detection results: (a).Original image; (b) Detected lines

Fig. 7 Evaluation graphs for horizontal and line detection.

4 Conclusions

This paper strives toward a novel methodology for automatic table detection in document images. The proposed methodology neither requires any training phase nor uses domain-specific heuristics, thus, resulting to an approach applied to a variety of document types. It builds upon several consequent stages that can be mainly identified to the following: (i) image pre-processing; (ii) horizontal and vertical line detection and (iii) table detection. Experimental results demonstrate the efficiency of the proposed method.

References

1. Zanibbi, R., Blostein, D., Cordy, J.: A survey of table recognition. International Journal of Document Analysis and Recogntion (IJDAR), vol. 7 (2004) 1-16
2. Zheng, Y., Liu, C., Ding, X., Pan, S.: Form Frame Line Detection with Directional Single-Connected Chain. Proc. of the 6th Int. Conf. on Doc. Anal. & Recognition (2001) 699-703
3. Neves, L. , Facon, J.: Methodology of Automatic extraction of Table-Form Cells. IEEE Proc. of the XIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'00) (2000) 15-21
4. Kieninger, T., Dengel, A.: Applying the T-Recs Table Recognition System to the Business Letter Domain. Proc. of the 6th International Conference on Document Analysis & Recognition, Seattle, (2001) 518-522
5. Cesari, F., Marinai, S., Sarti, L., Soda, G.: Trainable Table Location in Document Images. Proc. of the International Conference of Pattern Recognition, vol. 3 (2002) 236-240
6. Gatos, B., Pratikakis, I., Perantonis, S.J.: An adaptive binarisation technique for low quality historical documents. IARP Workshop on Document Analysis Systems (DAS2004), Lecture Notes in Computer Science (3163), (2004) 102-113
7. Yin, P.Y.: Skew detection and block classification of printed documents. Image and Vision Computing 19, (2001) 567-579
8. Perantonis, S.J., Gatos, B., Papamarkos, N.: Block decomposition and segmentation for fast Hough transform evaluation. Pattern Recognition, vol. 32(5) (1999) 811-824
9. Avila, B.T., Lins, R.D.: A new algorithm for removing noisy border from monochromatic documents. Proc. of the 2004 ACM Symp. on Applied Comp. (2004) 1219-1225
10. Antonacopoulos, A., Gatos, B., Karatzas, D.: ICDAR 2003 Page Segmentation Competition. Proc. of the 7th Int. Conf. on Document Analysis & Recognition (2003) 688-692
11. Zheng Yefeng homepage (2005): http://www.ece.umd.edu/~zhengyf/