Digital line segment detection for table reconstruction in document images

Phuc Ngo

Digital line segment detection for table reconstruction in document images

Phuc Ngo[1][0000−0002−7423−5932]

Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France
hoai-diem-phuc.ngo@loria.fr

Abstract. Table detection is often involved in many applications of document analysis as tables are frequently used to present structured information. In this context, we are interested in extracting table regions in document images. More precisely, we propose a method for table detection based on a recent edge line detector which is developed in the context of digital geometry and it allows to handle noisy document images. The extracted lines are then used to reconstruct the tables contained in the image. The method has been evaluated and compared to other state-of-the-art methods and shown a very competitive result.

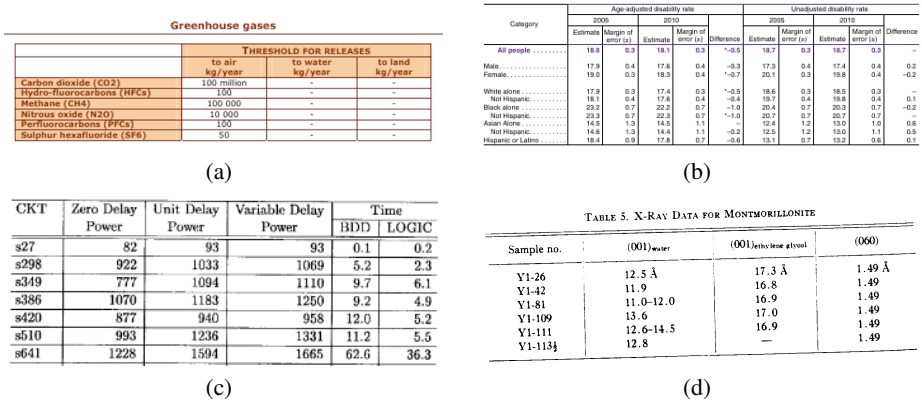
Keywords: Digital line / segment detection · Blurred segment · Adaptive directional scan · Materialized table extraction · Digital geometry

1 Introduction

The process of table analysis involves table detection and table structure recognition. The task of table detection aims to locate regions of image that contain tables using the bounding boxes, while table structure recognition involves the identification of row and column layout information for individual table cells. It is an important task in many document analysis applications. Thank to their compact form, tables are frequently used to summarized information in a structured and relational way. They are present in a large variety of documents such as reports, scientific papers, business documents, invoice, ...

This paper addresses the table detection in document images. It is a difficult problem due to the variety of layouts and formats of tables in images (see Fig. 1). Furthermore, when tables are correctly extracted from images, it allows to perform the recognition task more efficiently, and thus improve the table information extraction such as the text recognition within each cell and the interpretation of tabular data in the document.

In the context of table detection, several approaches have been proposed in the literature. One of the earliest works on identifying tabular regions in document images is of Watanabe *et al.* [27] in which the authors proposed to identify blocks enclosed by vertical and horizontal line segments. More precisely, line segments are first detected and corner points are subsequently determined. The method then exploits the connective relationships between the extracted corner points for the individual blocks using global and local tree structures. Following this idea, the methods in [6, 10, 12, 14] use parallel lines, the horizontal and vertical space as features to extract the table regions. Other conventional techniques for table analysis have been studied based on the textual



Greenhouse gases

	THRESHOLD FOR RELEASES		
	to air kg/year	to water kg/year	to land kg/year
Carbon dioxide (C02)	100 million	-	-
Hydro-fluorocarbons (HFCs)	100	-	-
Methane (CH4)	100 000	-	-
Nitrous oxide (N2O)	10 000	-	-
Perfluorocarbons (PFCs)	100	-	-
Sulphur hexafluoride (SF6)	50	-	-

(a)

Category	Age-adjusted disability rate					Unadjusted disability rate				
	2005		2010			2005		2010		
	Estimate	Margin of error (±)	Estimate	Margin of error (±)	Difference	Estimate	Margin of error (±)	Estimate	Margin of error (±)	Difference
All people	18.6	0.3	18.1	0.3	*–0.5	18.7	0.3	18.7	0.3	–
Male	17.9	0.4	17.6	0.4	–0.3	17.3	0.4	17.4	0.4	0.2
Female	19.0	0.3	18.3	0.4	*–0.7	20.1	0.3	19.8	0.4	–0.2
White alone	17.9	0.3	17.4	0.3	*–0.5	18.6	0.3	18.5	0.3	–
Not Hispanic	18.1	0.4	17.6	0.4	–0.4	19.7	0.4	19.8	0.4	0.1
Black alone	23.2	0.7	22.2	0.7	*–1.0	20.4	0.7	20.3	0.7	–0.2
Not Hispanic	23.3	0.7	22.3	0.7	*–1.0	20.7	0.7	20.7	0.7	–
Asian Alone	14.5	1.3	14.5	1.1	–	12.4	1.2	13.0	1.0	0.6
Not Hispanic	14.6	1.3	14.4	1.1	–0.2	12.5	1.2	13.0	1.1	0.5
Hispanic or Latino	18.4	0.9	17.8	0.7	–0.6	13.1	0.7	13.2	0.6	0.1

(b)

CKT	Zero Delay Power	Unit Delay Power	Variable Delay Power	Time BDD	Time LOGIC
s27	82	93	93	0.1	0.2
s298	922	1033	1069	5.2	2.3
s349	777	1094	1110	9.7	6.1
s386	1070	1183	1250	9.2	4.9
s420	877	940	958	12.0	5.2
s510	993	1236	1331	11.2	5.5
s641	1228	1594	1665	62.6	36.3

(c)

TABLE 5. X-RAY DATA FOR MONTMORILLONITE

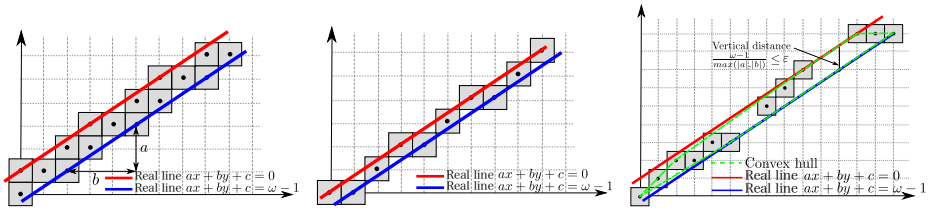
Sample no.	$(001)_{water}$	$(001)_{ethylene\ glycol}$	(060)
Y1-26	12.5 Å	17.3 Å	1.49 Å
Y1-42	11.9	16.8	1.49
Y1-81	11.0–12.0	16.9	1.49
Y1-109	13.6	17.0	1.49
Y1-111	12.6–14.5	16.9	1.49
Y1-113¾	12.8	—	1.49

(d)

Fig. 1: Examples of materialized tables in document images. Images extracted from ICDAR 2013 Table Competition [13] (a–b) and from UNLV Table Dataset [26] (c–d).

layout analysis of documents. We can mention the work of T. Kieninger [18] which relies on the detection of text block units for extracting the table cells. More precisely, the method starts with an arbitrary word as block seed, and recursively extends this block to all the words that interlace with their vertical neighbors. Then, it uses the gap information to prevent the table columns. Similarly, S. Mandal *et al.* in [21] supposed that tables are present in distinct columns and the spaces between the fields are larger than those between words in text lines. Then, the table extraction relies on the formation of word blob text lines and on finding the set of text lines that may form a table. In [25], the method is based on the layout analysis module of Tesseract OCR (Optical Character Recognition) to detect tables. More specifically, it uses the tab-stops to inform text aligns and to extract blocks composing the layout of a document. These blocks are then evaluated as table candidates using a specific strategy. Recently, the deep-learning approaches based on convolutional neural network (CNN) have been significantly investigated for table analysis. T. Kieninger and A. Dengel [28] can be considered as the pioneers to apply unsupervised learning method to the table detection using a bottom-up clustering of given word segments. After that, we have the supervised methods such as in [5, 11] that used Faster R-CNN based model for table detection, or [22, 23] used CNN to performed table detection and table structure recognition at the same time.

The present work aims at designing a method of table detection using image processing techniques. In particular, we are interested in *materialized tables* in documents. Such tables have rectangular form and are composed by straight line segments intersecting at right angle. They may not be closed but must contain at least two surrounded horizontal lines and inside vertical lines (see Fig. 1). For detecting these tables, the method relies on a recent line detector called *Fast Blurred Segment Detector* (FBSD) which is developed in the context of digital geometry. This detector is fast, accurate and robust to noise. It can be run in pure automatic mode, with quite few parameter settings. The extracted lines are then used as candidates to reconstruct the tables in document images. The method has been evaluated and compared to other state-of-the-art methods on different public datasets and shown a very competitive result.



(a) Standard segment (b) Naif segment (c) Blurred segment

Fig. 2: Examples of (a) a standard segment belongs to the 4-connected digital line $\mathcal{D}(2, -3, 3, 5)$, (b) a naif segment belongs to the 8-connected digital line $\mathcal{D}(2, -3, 0, 3)$ and (c) a blurred segment of thickness $\varepsilon = 1.5$ belongs to the digital line $\mathcal{D}(3, -4, 3, 7)$.

The paper is organized as follows. Sec. 2 recalls the main theoretical notions used in this work. Sec. 3 describes the FBSD detector for extracting straight line segments in images. The proposed method of table extraction is presented in Sec. 4. Then, Sec. 5 shows the experimental results, conclusion and future works are provided in Sec. 6.

2 Background Notions

We recall hereafter several notions of digital geometry [20] used in this work. We refer the reader to the given references for more details.

2.1 Digital Straight Line and Blurred Segment

Definition 1. *[24] A **digital straight line** of integer characteristics $(a, b, c, \omega) \in \mathbb{Z}^4$ with a and b relatively prime, denoted by $\mathcal{D}(a, b, c, \omega)$, is the set of digital points $p = (x, y) \in \mathbb{Z}^2$ satisfying the inequalities:*

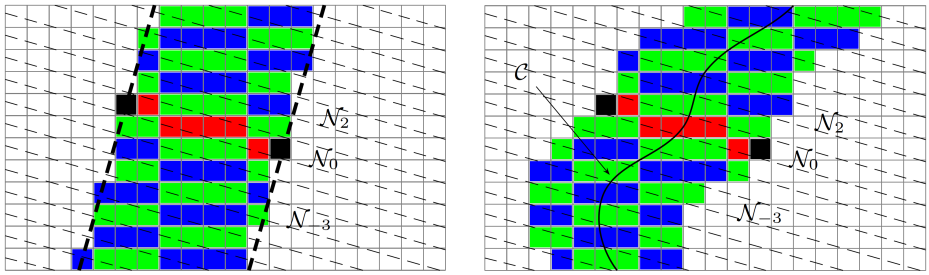
$$0 \leq ax + by + c < \omega \tag{1}$$

Hereafter, we note $\vec{V}(\mathcal{D}) = (-b, a)$ the director vector of $\mathcal{D}$, $w(\mathcal{D}) = \omega$ its arithmetical width, $h(\mathcal{D}) = -c$ its shift to origin, and $p(\mathcal{D}) = max(|a|, |b|)$ its period. The thickness $\mu = \frac{\omega - 1}{p(\mathcal{D})}$ of $\mathcal{D}$ is the minimum of the vertical and horizontal distances between lines $ax + by + c = 0$ and $ax + by + c = \omega - 1$. When $\omega = \max(|a|, |b|)$ then $\mathcal{D}$ is the narrowest 8-connected line and called *naif digital line*, and $\omega = |a| + |b|$ then $\mathcal{D}$ is 4-connected line and called *standard digital line*.

A *digital straight segment* is a finite subset of $\mathcal{D}$. Figure 2 (a–b) show examples of digital straight lines and segments.

Definition 2. *[16] A **blurred segment** $\mathcal{B}$ of assigned thickness ε is a sequence of points that are all covered by a digital straight line $\mathcal{D}$ of thickness $\mu \leq \varepsilon$. The covering digital line with minimal thickness is called the* optimal line *of $\mathcal{B}$.*

Example of blurred segment is given in Fig. 2 (c). In [16], it is shown that blurred segments can be detected in linear time by a recognition algorithm based on an incremental growth of the convex hull of added points.



(a) Static directional scan (b) Adaptive directional scan

Fig. 3: (a) A static directional scan with the start scan S_0 in red, odd scans in green, even scans in blue, bounds of scan lines $\mathcal{N}_i$ with dashed lines and bounds of scan strip $\mathcal{D}$ with bold dashed lines. (b) An adaptive directional scan in which the scan strip is dynamically fit to the curve C, and the scans are accordingly shifted to cover C.

2.2 Adaptive Directional Scan

From the notion of blurred segment, two notions of directional scan are proposed in [7, 17] for thick line detection in gray-level images.

Definition 3. *[17] A **static directional scan** is an ordered partition into scans S_i restricted to the grid domain $\mathcal{G} \subset \mathbb{Z}^2$ of a thick digital straight line $\mathcal{D}$, called* scan strip*. Each scan S_i is a segment of a naive line $\mathcal{N}_i$, called* scan line*, orthogonal to $\mathcal{D}$. The directional scan is defined as:*

$$DS = \left\{ S_i = \mathcal{D} \cap \mathcal{N}_i \cap \mathcal{G} \left| \begin{array}{l} \vec{V}(\mathcal{N}_i) \cdot \vec{V}(\mathcal{D}) = 0 \\ h(\mathcal{N}_i) = h(\mathcal{N}_{i-1}) + p(\mathcal{D}) \end{array} \right. \right\} \tag{2}$$

In this definition, $\vec{V}(\mathcal{N}_i) \cdot \vec{V}(\mathcal{D}) = 0$ expresses the orthogonality between the scan lines $\mathcal{N}_i$ and the scan strip $\mathcal{D}$. The shift $p(\mathcal{D})$ between successive scans $\mathcal{N}_{i-1}$ and $\mathcal{N}_i$ guarantees that all points of $\mathcal{D}$ are traversed only one time. The scans S_i can be iteratively parsed from the start scan S_0 to both ends (see Fig. 3 (a)).

An **adaptive directional scan** is a dynamical version of the directional scan with an on-line registration to a moving search direction. Compared to static directional scans where the scan strip remains fixed to the initial line $\mathcal{D}_0$, here the scan strip $\mathcal{D}_i$ follows a curve C to track while scan lines $\mathcal{N}_i$ remain fixed (see Fig. 3 (b)).

Definition 4. *[7] An **adaptive directional scan** is defined by:*

$$ADS = \left\{ S_i = \mathcal{D}_i \cap \mathcal{N}_i \cap \mathcal{G} \left| \begin{array}{l} \vec{V}(\mathcal{N}_i) \cdot \vec{V}(\mathcal{D}_0) = 0 \\ h(\mathcal{N}_i) = h(\mathcal{N}_{i-1}) + p(\mathcal{D}_0) \\ \mathcal{D}_i = \mathcal{D}(\widehat{C}_i, w(\mathcal{D}_0)), i > 0 \end{array} \right. \right\} \tag{3}$$

where $\widehat{C}_i$ is a triplet composed of the director vector (a_i, b_i) and the shift to origin c_i of an estimate at position i of the tangent to the curve C to track.

The obtained thick digital line is used to update the scan strips and lines. The last clause expresses the scan bounds update at iteration i.

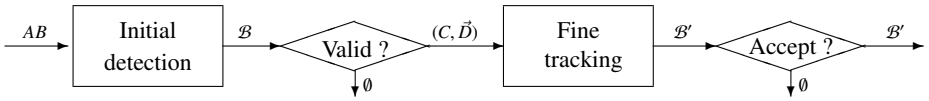


Fig. 4: Flowchart of the thick line segment detection process by FBSD [7].

3 FBSD: Fast Blurred Segment Detector

In this section, we describe briefly the efficient tool FBSD [7] to detect blurred segments from gray-level images based on discrete geometry notions presented previously. More precisely, the method examines the gradient image of the input for edge detection. Firstly, it uses gradient magnitude to select image points for the edge candidates. Then, it relies on gradient orientation to track edges and to detect the blurred segments. The obtained segments encode straight edges, and its thickness parameter characterizes the edge sharpness. In particular, a linear-time blurred segment recognition algorithm is applied to incrementally extend the segment and the adaptive directional scan allows to orientate the edge candidates in appropriate directions in gradient image.

The process for a single detection, given an input seed segment AB crossing the edges to be detected, is summarized in Fig. 4. It is composed of two steps. The initial detection consists in building and extending a rough blurred segment $\mathcal{B}$ of assigned thickness ε_0, based on points with highest gradient magnitude found in each scan of the directional scan defined by AB with a large tolerance to gradient direction. Validity tests, for small and sparse segments, are then applied to decide the detection pursuit.

In the second step of fine tracking, the final blurred segment $\mathcal{B}'$ is built and extended with points that correspond to local maxima of the gradient, ranked by magnitude order, and with gradient direction close to start point gradient direction. At this refinement step, the adaptive directional scan, defined by the found center position C and the rough blurred segment direction $\vec{D}$, is used in order to extend the segment in the appropriate direction. As soon as the thickness of the expanding blurred segment gets stable, a procedure is applied to focalize the search direction and to avoid further insertion of ouliers. Final length and sparsity thresholds of the output segment $\mathcal{B}'$ can be set accordingly. They are the only parameters of this local detector, together with the input assigned thickness ε_0.

By considering the prominent local maxima of the gradient magnitude under AB, the previous process also allows the detection of all the edges crossed by the seed segment AB, called *multi-detection*. Furthermore, in order to avoid multiple detections of the same edge, an occupancy mask is used to collect the dilated points of all the detected blurred segments, so that these points can not be used any more in the detection.

In [7], a full automatic detection of all straight lines in image is also proposed. More precisely, a line segment, that crosses the whole image, is swept in both directions, vertical then horizontal, from the center to the borders of the image. At each position, the multi-detection algorithm is run to collect all the blurred segments found under the sweeping segment. Then, small segments are rejected in order to avoid the formation of misaligned segments when the sweeping segment crosses an image edge near one of its ends. For more details of the FBSD detector, we refer the reader to the paper [7].

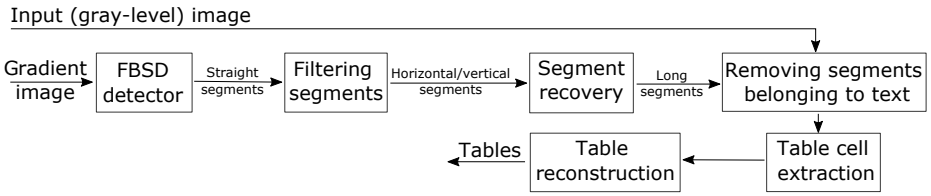


Fig. 5: Flowchart of the proposed method for table extraction from document images.

4 Main Table Extraction Framework

This section presents the proposed method for extracting tables from document images. In [14], a table is said *fully materialized* if it is composed and surrounded by straight line segments intersecting at right angle. In our work, we concern with tables of rectangular form which are also composed by straight line segments intersecting at right angle. However, they may not be closed, compared to fully materialized tables, but must contain at least two surrounded horizontal lines and inside vertical lines. We call them *materialized tables*. Such tables are quite frequent in a wide variety of documents such as invoices, scientific papers, commercial / administrative documents, question forms, ... Some examples of materialized tables are given in Fig. 1.

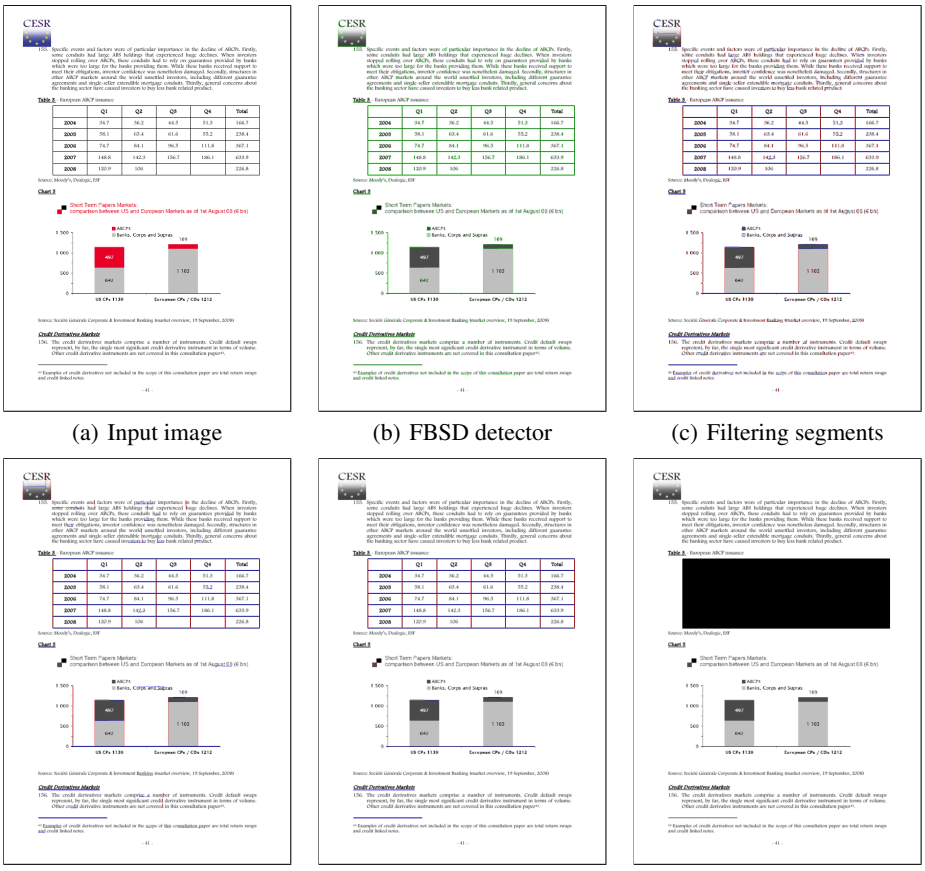
The proposed method relies on the edge line detector FBSD described in the previous section. The lines in document images are not all table edges, they can be part of graphics, images or separators. It should be mentioned that the acquisition noise during the sequential scanning / printing process may disrupt, damage or misalign the straight line segments (see Fig. 1 (c–d)) and it makes the task of table reconstruction and detection more difficult. This leads us to a proposal of line segment recovery before the reconstruction of tables from document images.

The automatic table extraction process is composed of 6 steps. These steps are detailed in the following. The overview of the whole process is given in Fig. 5 and illustrated in Fig. 6.

Line segment detection: This first step consists in extracting line segments in document images (see Fig. 6 (b)). For this, we use the FSBS detector [7]. These lines consist candidate segments of tables to be extracted and also other segments belonging to graphics, images, separators, texts, ...

Horizontal and vertical segment extraction: As we address only materialized tables which are composed of horizontal and vertical segments intersecting at right angle, we thus remove all segments which are not horizontal and vertical (see Fig. 6 (c)). A tolerance of $\alpha = \pm 5^\circ$ angle is applied for such segment selection due to impact of scanning procedure of documents.

Line segment recovery: In order to deal with broken lines, especially in degraded conditions and due to the acquisition noise, and to ensure a better stability of table detection, the extracted horizontal (resp. vertical) segments are prolonged and connected if they are close to each other (see Fig. 6 (d)). A distance threshold $\delta = 30$ pixels is used to test the closeness between two horizontal (resp. vertical) segments.



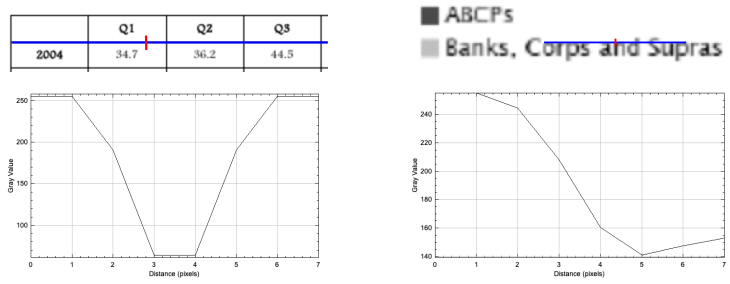
(a) Input image (b) FBSD detector (c) Filtering segments

(d) Long segments (e) Removing text segments (f) Output tables

Fig. 6: Different steps of the table extraction process: (a) input image, (b) straight segments detected by FBSD (green), (c) candidate horizontal (blue) and vertical (red) segments, (d) recovering nearby segments by extending and connecting horizontal (blue) and vertical (red) segments, (e) removing segments associated to text and (f) final result of table extraction (black box).

Suppression of segments belonging to text: The extracted segments from the previous step do not always belong to borders of cell tables. In particular, most of them are associated to text in document image. In order to eliminate such segments, we consider the original gray-level image and verify the line model of border cell tables. A local analysis around the extracted segment is performed. It can be observed that the pattern of the 1D gray intensity profile the segment of a cell border is typical with a bottom (of very low intensity) in the middle and high intensity on both sides[1], and it is not always

[1] A similar line model has been used in [14] for line verification and table extraction. Their model consists in separating the 1D intensity profile, from left to right, into 3 zones: the intensity should begin to increase then (potentially) stabilize and decrease.



(a) (b)

Fig. 7: Gray intensity profile (red segment) analysis along extracted segment (blue segment): (a) border cell table, and (b) text.

the case for segments associated to text (see Fig. 7). This profile pattern is repetitive along the segment associated to the border cell tables, but not in case of text. Therefore, by simply computing the ratio of the number of profile patterns corresponding to the border cell segment over the segment length, we can distinguish the segments associated to text. In our experiments, the ratio is set at 75% for eliminating text segments, and $w = 7$ pixels around the extracted segment is used for the local analysis of intensity profile. The result of this step is illustrated in Fig. 6 (e).

Table cell extraction: This step aims at detecting table cells before the final step of table reconstruction. The cell extraction is based on the vertical and horizontal segments limiting the table cells. Indeed, for the materialized tables, these cells are limited by four segments defining a rectangle. In case of open tables, *i.e.,* they are not bounded by vertical segments (see Fig. 1 (c–d)), the cells at extremities are surrounded by only three segments (two horizontal and one vertical segments), then we can close the cell by using the end points of the horizontal segments. By that way, we can eliminate, from the extracted segments, the separator lines which do not form a close rectangle.

Table reconstruction: Finally, a table is built as unions of all its cells (see Fig. 6 (f)). For this step, we compute the connected components from the extracted cells, and the corresponding table is obtained as the bounding box of each connected component. A verification of table size is performed at the end to remove small bounding boxes related to noise or other elements which may not be the tables. For this, we keep only the bounding boxes of size bigger than 1% of image size.

5 Experimental Results

This section presents results of the proposed method for table extraction from document images. In the experimental stage, the proposed approach is validated through comparisons with other recent methods in the literature: S. Faisal and S. Ray [25], H. Alhéritière *et al.* [14], D. Prasad *et al.* [23] and M. Li *et al.* [22]. In [25], the authors proposed a method based on components of the layout analysis module of Tesseract to locate tables in document images. In [14], the method relies on the local Radon Transform for line extraction and then the pattern recognition techniques for reconstructing

Dataset	Method	P	R	F_1
ICDAR 2013 Dataset	Faisal *et al.* [25][2]	0.25	0.33	0.28
	Alhéritière *et al.* [14][3]	0.98	0.83	0.90
	Prasad *et al.* [23]	**1**	**1**	**1**
	M. Li *et al.* [22]	0.97	0.80	0.88
	Our proposal	0.91	0.85	0.88
UNLV Table Dataset	Faisal *et al.* [25]	0.86	0.79	0.82
	Alhéritière *et al.* [14]	0.73	0.66	0.70
	M. Li *et al.* [22]	**0.92**	**0.96**	**0.94**
	Our proposal	0.83	0.82	0.83
Marmot Dataset	M. Li *et al.* [22]	0.77	**0.98**	0.86
	Our proposal	**0.93**	0.81	**0.87**

Table 1: Evaluation results of table detection in document images on different datasets.

tables from the extracted lines. [23] and [22] are deep-learning approaches based on convolution networks which allow to detect the regions of tables and recognize the structural body cells from the detected tables at the same time. For more details about these methods, we refer the reader to the given references.

Comparison results are run on three publicly available datasets for table detection: ICDAR 2013 Table Competition [13], UNLV Table Dataset [26] and Marmot Dataset [8]. More precisely, the ICDAR 2013 [1] contains 128 documents with a total of 150 tables: 75 tables in 27 excerpts from the EU and 75 tables in 40 excerpts from the US Government. The UNLV Table Dataset [4] contains 427 examples in scanned image format from a variety of sources ranging from technical reports and business letters to newspapers and magazines. The Marmot Dataset [2] has 2000 documents mainly from research papers composed of both Chinese and English pages, in which 1000 pages containing at least one table, while the other 1000 pages do not contain tables, but have complex layout where some page components may be mistakenly recognized as tables, such as matrices and figures. All these datasets are provided with information, in XML format, of table structure ground-truth data such as bounding box coordinates, rows, columns, cells, ...

To evaluate the performance of the table detection algorithms, we use the standard evaluation metrics in the literature [15, 19]: precision (P), recall (R) and F_1-measure (F_1) which is an harmonic mean of P and R. The metrics are computed by summing up the overlapping area of the obtained result and the ground-truth. More precisely, let D be the set of pixels of the extracted tables, and G of the ground-truth. Then, the considered measure are computed as:

$$P = \frac{D \cap G}{D}, \quad R = \frac{D \cap G}{G}, \quad F_1 = \frac{2 * R * P}{R + P} \tag{4}$$

The results are reported in Tab. 1 on the overall average score of each dataset. We observe that our method archives well-scored and competitive results comparing to the others on the considered datasets. More precisely, for the ICDAR 2013 dataset, [23]

[2] Results extracted from [14].
[3] Results on a subset of ICDAR 2013 containing 88 tables in [14].

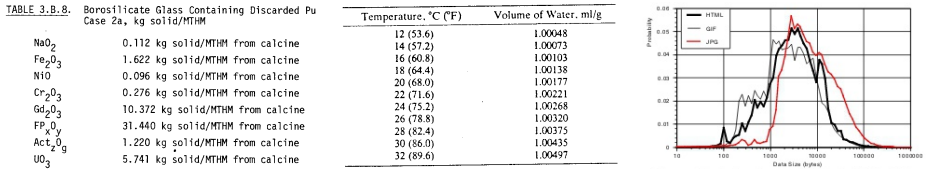


TABLE 3.B.8. Borosilicate Glass Containing Discarded Pu Case 2a, kg solid/MTHM

NaO_2	0.112 kg solid/MTHM from calcine
Fe_2O_3	1.622 kg solid/MTHM from calcine
NiO	0.096 kg solid/MTHM from calcine
Cr_2O_3	0.276 kg solid/MTHM from calcine
Gd_2O_3	10.372 kg solid/MTHM from calcine
FP_xO_y	31.440 kg solid/MTHM from calcine
Act_zO_g	1.220 kg solid/MTHM from calcine
UO_3	5.741 kg solid/MTHM from calcine

Temperature, °C (°F)	Volume of Water, ml/g
12 (53.6)	1.00048
14 (57.2)	1.00073
16 (60.8)	1.00103
18 (64.4)	1.00138
20 (68.0)	1.00177
22 (71.6)	1.00221
24 (75.2)	1.00268
26 (78.8)	1.00320
28 (82.4)	1.00375
30 (86.0)	1.00435
32 (89.6)	1.00497

Fig. 8: Limit cases of our proposal. Images from UNLV [26] and Marmot [8] datasets.

performed best at the absolute score of 100% on the evaluation measures, and our proposal is as good as [22]. In case of the UNLV Table dataset, [22] provides the best score and our method comes in second place. This can be explained by the very low quality of scanned document images, and also this dataset contains no materialized tables (see Fig. 8). Finally, on the Marmot dataset, the proposed method has the best precision and F_1 measures, but less good than [22] on recall. This lower recall can be explained by the non detection of materialized tables and the false detection of graphics in this dataset. Though, it should be mentioned that this dataset contains high quality images.

Further comparisons with other methods in the literature are also available in the referenced papers [8, 22, 23, 25].

6 Conclusion

In this paper, we presented a method for table extraction from document images based on notions of digital geometry. The method relies on the FBSD detector [7] to detect the straight line segments in image and then reconstruct the tables from the extracted lines. This detector is fast, accurate and robust to noise. In particular, it allows to handle noisy document images. Indeed, the sequential printing and scanning of a document may deteriorate its contents, including the table structures such as disrupting table lines. The proposed approach has been evaluated and compared with other methods on different table datasets. The results are encouraging and competitive to the existing methods. The source code of the proposed method, based on OpenCV library [3], is available at the *GitHub* repository: `https://github.com/ngophuc/TableExtraction`.

In the field of document analysis and interpretation of tabular data in document images, the table structure recognition [9,28] is also an active research topic. It involves identifying the rows, columns, individual table cells. In an advantage context, it is to retrieve the textual content of the cells. It is widely used in image captioning, video description, and many other applications. In this context, the proposed method can be helpful as a first step of table detection before extracting other information. On the other hand, we can use the method to address the need for generating data for the machine learning approaches which need many data for the training process.

As mentioned previously, we address the materialized tables in this work. However, in the considered datasets, there are tables not for this case study, and also graphics or matrices that our algorithm mistakenly detects as tables, see Fig. 8 for some examples. In future, we would like to extend the method for more general cases of tables and to conduct more comparisons with the state-of-the-art methods on more data.

References

1. Icdar 2013 table competition dataset, https://www.tamirhassan.com/html/competition.html
2. Marmot dataset, https://www.icst.pku.edu.cn/cpdp/sjzy/index.htm
3. OpenCV: Open computer vision, https://opencv.org/
4. Unlv table dataset, https://github.com/tesseract-ocr/
5. Arif, S., Shafait, F.: Table detection in document images using foreground and background features. In: 2018 Digital Image Computing: Techniques and Applications (DICTA). pp. 1–8 (2018). https://doi.org/10.1109/DICTA.2018.8615795
6. Cesarini, F., Marinai, S., Sarti, L., Soda, G.: Trainable table location in document images. In: 2002 International Conference on Pattern Recognition. vol. 3, pp. 236–240 vol.3 (2002). https://doi.org/10.1109/ICPR.2002.1047838
7. Even, P., Ngo, P., Kerautret, B.: Thick line segment detection with fast directional tracking. In: Proc. of 20th International Conference on Image Analysis and Processing. LNCS, vol. 11752, pp. 159–170. Springer, Trento, Italy (September 9–13 2019). https://doi.org/10.1007/978-3-030-30645-8_15
8. Fang, J., Tao, X., Tang, Z., Qiu, R., Liu, Y.: Dataset, ground-truth and performance metrics for table detection evaluation. In: 2012 10th IAPR International Workshop on Document Analysis Systems. pp. 445–449 (2012). https://doi.org/10.1109/DAS.2012.29
9. Farrukh, W., Foncubierta-Rodriguez, A., Ciubotaru, A.N., Jaume, G., Bejas, C., Goksel, O., Gabrani, M.: Interpreting data from scanned tables. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 02, pp. 5–6 (2017). https://doi.org/10.1109/ICDAR.2017.250
10. Gatos, B., Danatsas, D., Pratikakis, I., Perantonis, S.: Automatic table detection in document images. vol. 3686, pp. 609–618 (08 2005). https://doi.org/10.1007/11551188_67
11. Gilani, A., Qasim, S.R., Malik, I., Shafait, F.: Table detection using deep learning. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 01, pp. 771–776 (2017). https://doi.org/10.1109/ICDAR.2017.131
12. Green, E., Krishnamoorthy, M.: Model-based analysis of printed tables. pp. 214–217 (01 1995). https://doi.org/10.1109/ICDAR.1995.598979
13. Göbel, M., Hassan, T., Oro, E., Orsi, G.: Icdar 2013 table competition. In: 2013 12th International Conference on Document Analysis and Recognition. pp. 1449–1453 (2013). https://doi.org/10.1109/ICDAR.2013.292
14. Héloïse, A., Walid, A., Florence, C., Camille, K., Jean-Marc, O., Nicole, V.: Straight line reconstruction for fully materialized table extraction in degraded document images. In: Couprie, M., Cousty, J., Kenmochi, Y., Mustafa, N.H. (eds.) Discrete Geometry for Computer Imagery - 21st IAPR International Conference (DGCI). Lecture Notes in Computer Science, vol. 11414, pp. 317–329. Springer (2019). https://doi.org/10.1007/978-3-030-14085-4_25
15. Hu, J., Kashi, R., Lopresti, D., Wilfong, G.: Medium-independent table detection (12 1999)
16. Isabelle, D.R., Fabien, F., Jocelyne, R.D.: Optimal blurred segments decomposition of noisy shapes in linear time. Computers & Graphics **30**(1), 30–36 (2006)
17. Kerautret, B., Even, P.: Blurred segments in gray level images for interactive line extraction. In: Wiederhold, P., Barneva, R.P. (eds.) Proc. of Int. Workshop on Combinatorial Image Analysis. LNCS, vol. 5852, pp. 176–186 (2009)
18. Kieninger, T.: Table structure recognition based on robust block segmentation. pp. 22–32 (1998)
19. Kieninger, T., Dengel, A.: An approach towards benchmarking of table structure recognition results. pp. 1232–1236 (08 2005). https://doi.org/10.1109/ICDAR.2005.47

20. Klette, R., Rosenfeld, A.: Digital geometry – Geometric methods for digital picture analysis. Morgan Kaufmann (2004)
21. Mandal, S., Chowdhury, S., Das, A., Chanda, B.: Simple and effective table detection system from document images. International Journal on Document Analysis and Recognition **8**, 172–182 (06 2006). https://doi.org/10.1007/s10032-005-0006-5
22. Minghao, L., Lei, C., Shaohan, H., Furu, W., Ming, Z., Zhoujun, L.: Tablebank: A benchmark dataset for table detection and recognition (2019)
23. Prasad, D., Gadpal, A., Kapadni, K., Visave, M., Sultanpure, K.: Cascadetabnet: An approach for end to end table detection and structure recognition from image-based documents (2020)
24. Reveillès, J.P.: Géométrie discrète, calcul en nombres entiers et algorithmique. Thèse d'état, Université Strasbourg 1 (1991)
25. Shafait, F., Smith, R.: Table detection in heterogeneous documents. pp. 65–72. DAS '10, Association for Computing Machinery, New York, NY, USA (2010). https://doi.org/10.1145/1815330.1815339
26. Shahab, A., Shafait, F., Kieninger, T., Dengel, A.: An open approach towards the benchmarking of table structure recognition systems. In: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems. p. 113–120. DAS '10, Association for Computing Machinery, New York, NY, USA (2010). https://doi.org/10.1145/1815330.1815345
27. T., W., H., N., Q., L., N., S.: Structure analysis of table-form documents on the basis of the recognition of vertical and horizontal line segments. In: Proceedings of International Conference on Document Analysis and Recognition (ICDAR '91). pp. 638–646 (1991)
28. Thomas, K., Andreas, D.: The t-recs table recognition and analysis system. vol. 1655, pp. 255–269 (11 1998). https://doi.org/10.1007/3-540-48172-9_21