PRIP-TR-125

Towards a Common Evaluation Strategy for Table Structure Recognition Algorithms

Tamir Hassan

Pattern
Recognition &
Image
Processing
Group

Institute of
Computer Aided Automation

Technical Report

Pattern Recognition and Image Processing Group
Institute of Computer Aided Automation
Vienna University of Technology
Favoritenstr. 9/183-2
A-1040 Vienna AUSTRIA
Phone: +43 (1) 58801-18351
Fax: +43 (1) 58801-18392
E-mail: tamir@tamirhassan.com
URL: http://www.prip.tuwien.ac.at/

PRIP-TR-125 April 2010

Towards a Common Evaluation Strategy for Table Structure Recognition Algorithms

Tamir Hassan

Abstract

A number of methods for evaluating table structure recognition systems have been proposed in the literature, which have been used successfully for automatic and manual optimization of their respective algorithms. Unfortunately, the lack of standard, ground-truthed datasets coupled with the ambiguous nature of how humans interpret tabular data has made it difficult to compare the obtained results between different systems developed by different research groups.

With reference to these approaches, we describe our experiences in comparing our algorithm for table detection and structure recognition with another recently published system using a freely available dataset of 75 PDF documents. Based on examples from this dataset, we define several classes of errors and propose how they can be treated consistently to eliminate ambiguities and ensure the repeatability of the results and their comparability between different systems from different research groups.

Contents

1 Introduction

In the OCR domain, an active research field in the previous 30 years, a number of ground-truthed datasets have been made available to researchers for the sole purpose of creating experimental results to enable different systems and approaches from various research groups to be compared with each other. In contrast, in the field of table structure recognition, which is still developing, no such dataset exists, particularly with respect to PDF documents. Although the well-known *University of Washington* datasets do include ground-truthed table areas within the document, they do not include any information on substructures such as rows and columns. Furthermore, the data is in scanned bitmap format and not in PDF.

In this paper, we describe our efforts in comparing our system for table structure recognition in PDF documents [2] to another academic approach, the *PDF-TREX* system by Ruffolo and Oro [8]. We are very grateful to the authors of PDF-TREX for providing us with a dataset of 75 documents and the output of their system on their dataset, which we have used to compare both systems. All the examples in this paper are from this dataset. This dataset has since been extended to 100 documents[1], and has recently been made freely available on the Internet [9].

The biggest hurdle that we encountered was how to consistently evaluate the various types of structure recognition error (split cell, merged cell, etc.) that occurred. In Section 2, we describe previous approaches to evaluating such errors and the problems that we encountered. In Section 3 we propose a classification methodology for each type of error that we encountered, and how it could be consistently evaluated in the future. We hope that this represents a step towards creating common, repeatable experimental results that can be compared between different systems from different research groups.

We also encountered further difficulties in ground truthing of the dataset (Section 5) and in aggregation of the results for each cell and each table to create a set of figures for the complete dataset (Section 6). Finally, Section 7 presents the results of our evaluation strategy on both systems and Section 8 concludes the paper.

[1] The 75 documents used for our comparison correspond to the following documents in the publicly available dataset: 1–12, 14–16, 18–23, 25–34, 37, 38, 40–42, 45–58, 60, 61, 63, 66–69, 71, 72, 75, 79–81, 83, 84, 86, 87, 89, 90, 92–98

2 Structure recognition issues

A common way to generate numerical values for the performance of table structure recognition algorithms is to borrow the notions of *recall* and *precision* from the information retrieval field [8, 10, 6, 7]. The PDF-TREX system was evaluated in this way, and separate figures for table areas and table cells were generated. The usual definitions of these measures are as follows:

$$\text{Recall} = \frac{\text{number of correctly retrieved data items}}{\text{total number of data items in dataset}}$$

$$\text{Precision} = \frac{\text{number of correctly retrieved data items}}{\text{total no. of retrieved data items}}$$

Essentially, recall measures the proportion of data that has been found correctly without regard to false positives, whereas precision is a measure of how good the algorithm is at avoiding false positives, without regard to recall. Many algorithms can be fine tuned to maximize recall at the expense of precision, and vice versa; and our algorithm is of no exception. The *F-measure*, which is defined as the harmonic mean of precision and recall, is often used as a single-figure measure of the ability of the system.

The biggest problem that arises with this approach is the not unambiguous interpretation of the term *correctly retrieved* in this context. For example, let us consider the simple case where a cell is erroneously split into two cells by the system. Note that precision is usually defined as the proportion of data items that have been *correctly retrieved.* Has the data in the split cell been "correctly retrieved" and do we therefore count these two cells as one true positive and one false positive, or as one false negative and one false positive?

We found that the evaluation strategies used by Ruffolo and Oro [8] and by Kieninger and Dengel [6, 7] would class at least one cell resulting from the split as having been *correctly retrieved*, even if the data was only partly retrieved (in the latter system, the *best match* according to the sub-objects is found and evaluated as being correct). The remaining cells of the split are classified as false positives, which results in this error affecting overall precision, but not recall.

This simple example highlights the problems with using such a simple model to represent errors in recognizing more complex structures. In our system, table cells may span multiple columns or consist of several lines of text. We therefore need to not only deal with split and merged cells, but with incorrectly detected colspans, for example. How should such an error be evaluated in terms of false and true positives?

Unfortunately, most previous publications in table structure recognition do not describe their evaluation strategy in sufficient detail to enable it to be reproduced precisely by a different team of researchers, in order to obtain a fair comparison of both systems.

In Section 3, we provide a classification of all structure recognition errors that we encountered in comparing our system against the PDF-TREX dataset and how they were evaluated. We hope that this provides a step towards a common method for evaluating table structure recognition results so that they can be directly compared between systems from different research groups.

3 A classification scheme for structure recognition errors

As described in the previous section, the relatively simple model of true positives, false positives and missed cells (true negatives) is not expressive enough to fully express the various types of errors that can occur. We have therefore defined a larger number of categories, as shown in Table 1, to explicitly represent the majority of errors that can occur. This table also shows whether the category was evaluated as a true positive, false positive, true positive or false negative in our final numerical results. After having calculated our initial results, we decided to create a second set of figures that better represented their usefulness for our application (data extraction), by reclassifying certain true positive categories as false positives.

In our results, the first occurrence of a split object was given the classification split and the extra occurrences arising from the split were classed as extra. By defining the classification split as a true positive, we obtain a similar evaluation metric to that used in [6, 8].

An important criterion in evaluating table structure recognition errors is the gravity of the error itself, and not just the number of cells that are affected. This may depend on the particular application. Let us consider a further example where a single cell is split horizontally, causing an otherwise blank column to be added in between the data columns in a table. A data extraction algorithm that locates the cells based on their headings will still continue to function correctly for the other cells, as the data cells remain correctly aligned. We therefore introduced two sub-classifications for non-empty split cells: split full and split data. In the former, the textual data within the cell is not split; only extra (false positive) blank cells result; in the latter, the textual data itself is split across several cells. When calculating

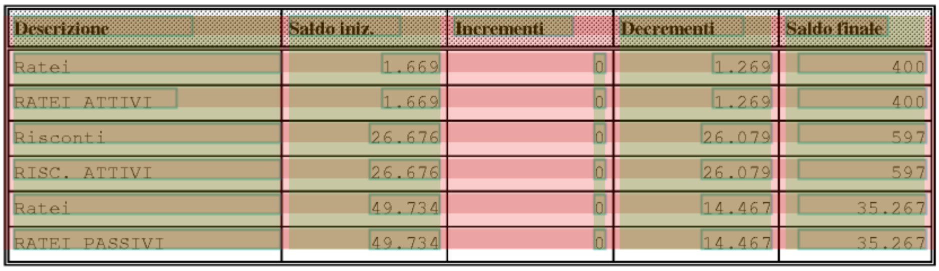
Table areas		**Data cells**		**Blank cells**	
Found correctly Data cells found Partially found	TP TP TP/FP	Found correctly	TP	Found correctly	TP
Split table	TP/FP	Split full Split data	TP TP/FP	Split blank	TP
Extra table	FP	Extra data	FP	Extra blank	FP
Incorrect table	FP	Incorrect data	FP	Incorrect blank	FP
Merged into surroundings Merged	TP TP/FP	Merged	TP/FP		
Not recognized	TN	Not recognized	TN	Not recognized	TN

Table 1: Classifications for table areas, data cells and blank cells

our results, we first classified both types of split cells as true positives. We then calculated a second set of totals by reclassifying split data cells as false positives, which we believe better reflects the usability of the result for our application.

A further question is whether blank cells should be counted at all. Whereas Ruffolo and Oro's evaluation strategy included blank cells, the strategy employed by Kieninger and Dengel appeared not to. As most data extraction applications only use the data cells, results which do not include blank cells in their totals could be seen as being more meaningful. Furthermore, for non-ruled tables, it is not always clear how many "blank" cells they contain, particularly in the case of cells along the edge of the table. In our case, we assumed each table to be rectangular in shape (according to our model), and represented any empty spaces along the table boundary as blank cells. For each set of results, we generated two sets of totals: one including both blank and data cells, and one excluding the blank cells.

Regarding table areas, it was noted that, in the PDF-TREX result set, even partly detected table areas counted towards the recall score. Therefore, we first chose to classify partially found tables as true positives. Therefore, such an error is only penalized by affecting the cell recall figure. The numbers of fully and partially found table areas were counted separately. A common error that occurred with many table areas was that all the data cells were found, but the heading cells, which were located some distance from the table body, were not. For data extraction purposes, such a result would be adequate, as it would still be possible to extract all the data from the table. We therefore introduced a further classification, data cells found. A similar situation occurred with the classification merged into surroundings,



Descrizione	Saldo iniz.	Incrementi	Decrementi	Saldo finale
Ratei	1.669	0	1.269	400
RATEI ATTIVI	1.669	0	1.269	400
Risconti	26.676	0	26.079	597
RISC. ATTIVI	26.676	0	26.079	597
Ratei	49.734	0	14.467	35.267
RATEI PASSIVI	49.734	0	14.467	35.267

Descrizione	Saldo indiz.	Incrementi		Decrementi	Saldo finale
Ratei	1.669		0	1.269	400
RATEI ATTIVI	1.669		0	1.269	400
Risconti	26.676		0	26.079	597
RISC. ATTIVI	26.676		0	26.079	597
Ratei	49.374		0	14.467	35.267
RATEI PASSIVI	49.374		0	14.467	35.267

Figure 1: Example of a table with a split column. The view from our interface is shown above; the resulting HTML table below

where tables were typically merged with neighbouring lines or text above or below, but it was still possible to extract all the data from the table.

The complete list of classifications that we used is shown in Table 1. As well as split data cells, we also chose to reclassify certain other classifications for our second set of totals as false positives to correspond to a more strict interpretation of the data items having been *correctly retrieved* and better reflect the usability of the result.

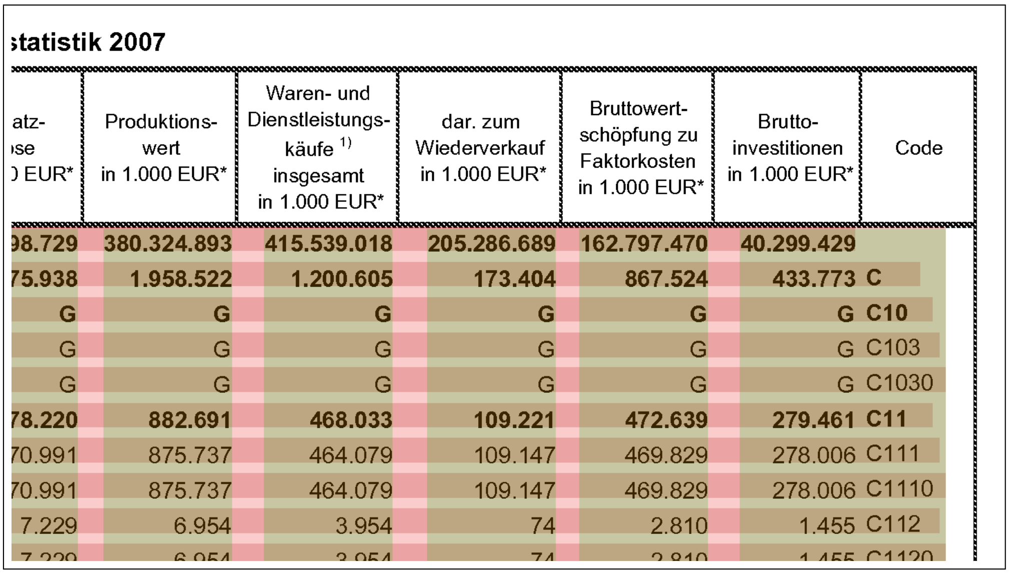
4 Types of errors

This section lists the various recognition errors that were encountered during evaluation of our table structure recognition algorithm and PDF-TREX and how they were evaluated, i.e. which cells were given which classifications. The totals for both systems are given in Table 2.

4.1 Cell errors

4.1.1 Splitting errors

1. single column in table is detected as two separate columns (see the example in Figure 1):



statistik 2007

atz- ıse) EUR*	Produktions- wert in 1.000 EUR*	Waren- und Dienstleistungs- käufe 1) insgesamt in 1.000 EUR*	dar. zum Wiederverkauf in 1.000 EUR*	Bruttowert- schöpfung zu Faktorkosten in 1.000 EUR*	Brutto- investitionen in 1.000 EUR*	Code
98.729	**380.324.893**	**415.539.018**	**205.286.689**	**162.797.470**	**40.299.429**	
75.938	**1.958.522**	**1.200.605**	**173.404**	**867.524**	**433.773**	**C**
G	**G**	**G**	**G**	**G**	**G**	**C10**
G	G	G	G	G	G	C103
G	G	G	G	G	G	C1030
78.220	**882.691**	**468.033**	**109.221**	**472.639**	**279.461**	**C11**
70.991	875.737	464.079	109.147	469.829	278.006	C111
70.991	875.737	464.079	109.147	469.829	278.006	C1110
7.229	6.954	3.954	74	2.810	1.455	C112
7.229	6.954	3.954	74	2.810	1.455	C1120

Figure 2: Example of a table with a merged column

(a) content of cell is not split; additional blank cell is introduced

- the cell is classified as split full; the resultant blank cell as extra blank

(b) content of cell is split into two (or more) cells

- the original cell is classified as split data; the resultant additional cell(s) as extra data

2. single, multi-line row is erroneously split into its constituent lines:

   - the original cell is classified as split full or split data, depending on whether the textual data has been split; the resultant additional cell(s) as extra data or extra empty

3. cell spanning several rows is not detected and split into individual rows:

   - the original cell is classified as split full or split data, depending on whether the textual data has been split; the resultant additional cell(s) as extra data or extra empty

4. cell (e.g. a heading) spanning several columns is not detected and split into its individual columns

- the original cell is classified as split full or split data, depending on whether the textual data has been split; the resultant additional cell(s) as extra data or extra empty

4.1.2 Merging errors

1. horizontal merging of cells in adjacent columns (e.g. due to insufficient whitespace between them, as in the example in Figure 2):

   (a) one or more cells detected as spanning; column structure remains in other rows

   - the spanning cell is classified once as merged; the remaining cells within it as not recognized; the remaining cells in the column are detected correctly in this case

   (b) no cells detected as spanning, i.e. all cells in column are merged and the entire column disappears

   - the spanning cells are classified once as merged; the remaining cells within them as not recognized; all remaining data and blank cells in the missing column are also classified as not recognized

2. merging of adjacent rows (i.e. 2 rows are seen as one multi-line row)

   - each resulting (incorrect) cell is classified as merged

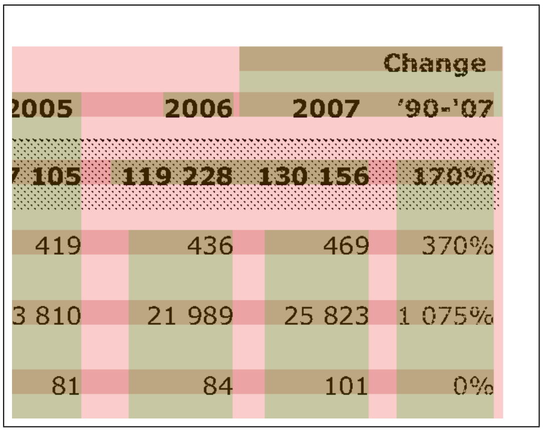
4.1.3 Other errors

1. cells are split in one direction and merged in another (see the example in Figure 3)

   - the horizontal error takes priority; in this example, the cell is classified as a single merged cell. Extra cells resulting from the split are still counted as normal

2. an entire non-spanning column of a table is seen as spanning several columns, except for a few individual cells, which do not span the entire width of the column

   - in this case, the spanning cells are classified as having been correctly detected; the non-spanning cells are classified as split full, and the resulting empty cells as extra blank



			Change
2005	2006	2007	'90-'07
7 105	119 228	130 156	170%
419	436	469	370%
3 810	21 989	25 823	1 075%
81	84	101	0%

Figure 3: Example of cells which are split in one direction and merged in another.

3. cells, which fall within the rectangular boundary of the table, are not recognized as belonging to the table (e.g. in Figure 4, they lie on the edge of the table and, due to the text being in a different font size, have been detected as surrounding text)

    - these cells are classified as not recognized; any empty cells in their place are incorrect empty

4.2 Table boundary errors

1. additional lines or columns are detected, outside of the table's actual boundary:

    - table is classified as merged into surroundings; the additional cells as incorrect data or incorrect empty

2. extra lines or other data is added to a cell along the edge of a table (but no extra cells are added to the table outside its boundary)

    - these cells are classified as merged; other cells in the row or column are unaffected; the table area classification is also unaffected (i.e., if no other errors are present, it is classified as found correctly)

3. lines or columns, which are part of the table, are not detected:

    - these are classified as not recognized

4. single table is split up into two or more tables across the data cells (see Figure 5)

    - the first table is classified as split; the resulting additional tables as extra table
    - all cells within the split tables belonging to the original full table are classified as normal, even if their respective colums or rows have been split across several tables; cells that have not been detected (e.g. between two split tables) are classified as not recognized

5. single table is split, but only across the heading/access cells (i.e. the heading cells are separated from the data; all data cells remain together)

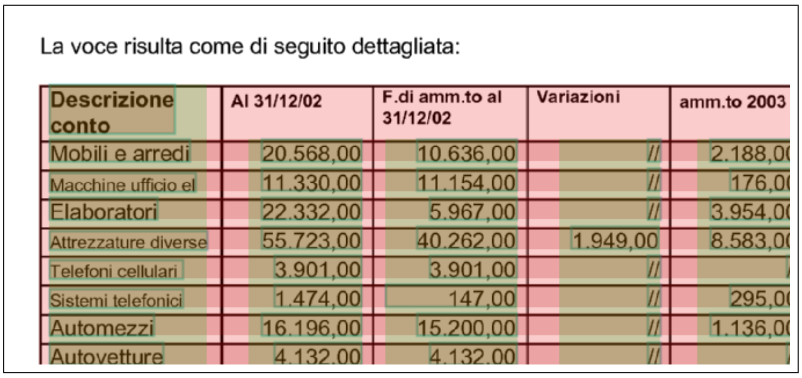
    - the table containing all data cells is classified as data cells found; the resulting additional tables as extra table
    - all cells within the split tables belonging to the original full table are classified as normal, even if their respective colums or rows have been split across several tables; cells that have not been detected (e.g. between two split tables) are classified as not recognized

6. neighbouring tables are merged; rows and/or columns align with each other

    - the first table is classified as merged; the resulting additional tables as not recognized. Cells from both original tables are classified normally

7. two horizontally neighbouring tables (or sub-tables) are merged and rows do not align (see Figure 6)

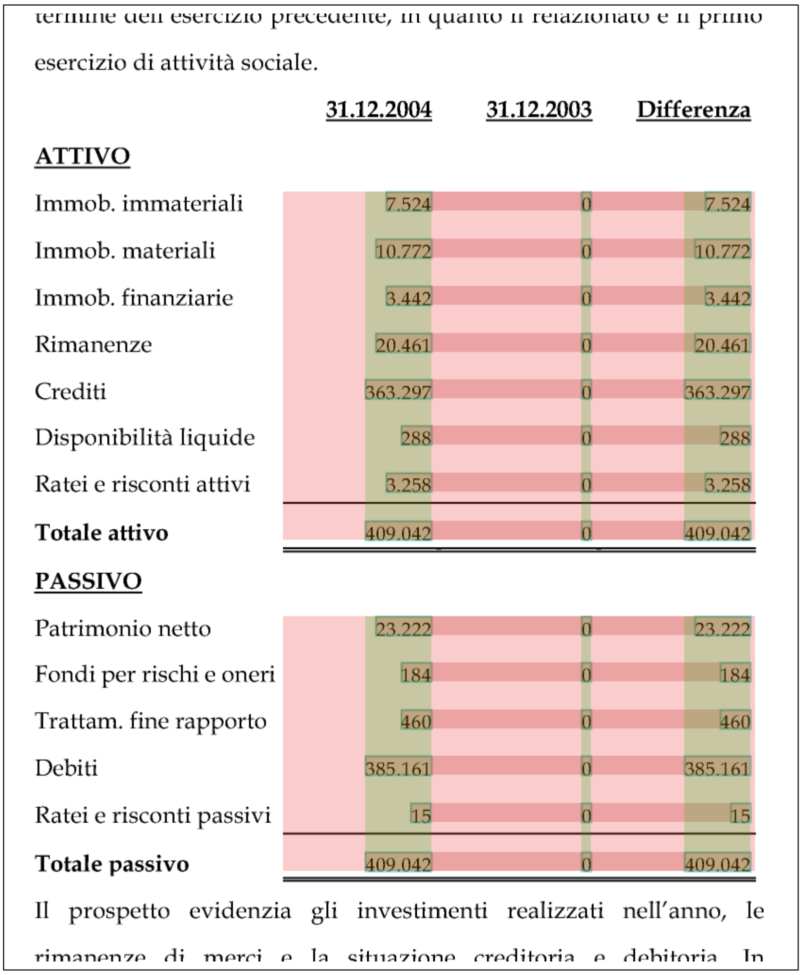
    - these tables may appear to be part of one large table. But, as their rows do not align with each other, it was decided to interpret these tables as separate tables. Therefore, in this example, the first is classified as merged; the second as not recognized. Cells in both tables are classified normally, although it is worth noting that a large number of incorrect blank rows and merged cells result as a result of the merge



La voce risulta come di seguito dettagliata:

Descrizione conto	Al 31/12/02	F.di amm.to al 31/12/02	Variazioni	amm.to 2003
Mobili e arredi	20.568,00	10.636,00	//	2.188,0
Macchine ufficio el	11.330,00	11.154,00	//	176,0
Elaboratori	22.332,00	5.967,00	//	3.954,0
Attrezzature diverse	55.723,00	40.262,00	1.949,00	8.583,0
Telefoni cellulari	3.901,00	3.901,00	//	/
Sistemi telefonici	1.474,00	147,00	//	295,0
Automezzi	16.196,00	15.200,00	//	1.136,0
Autovetture	4.132,00	4.132,00	//	/

Figure 4: Example of missing cells within a table boundary.

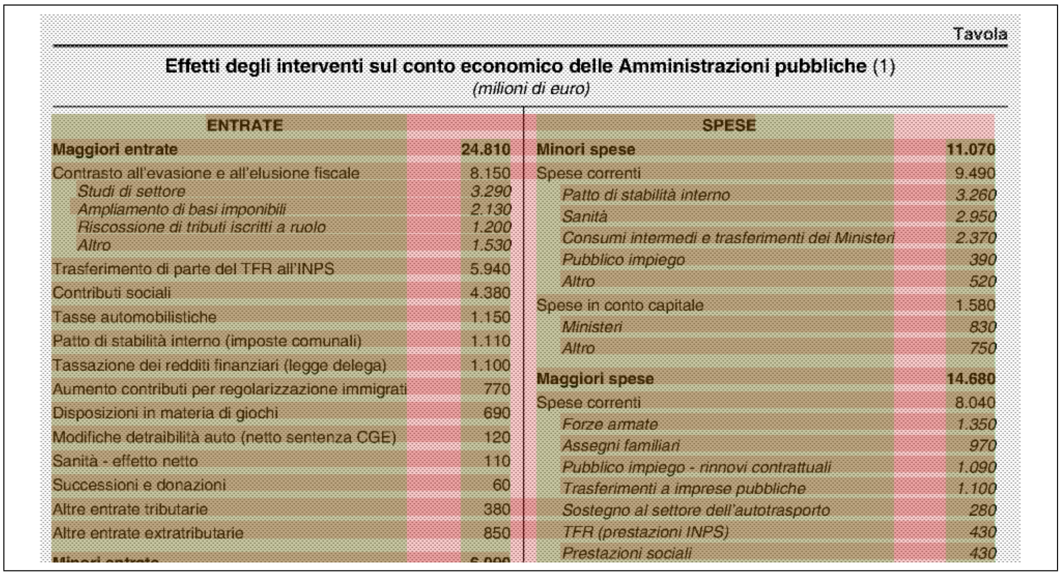


termine dell'esercizio precedente, in quanto il relazionato è il primo esercizio di attività sociale.

	31.12.2004	31.12.2003	Differenza
ATTIVO			
Immob. immateriali	7.524	0	7.524
Immob. materiali	10.772	0	10.772
Immob. finanziarie	3.442	0	3.442
Rimanenze	20.461	0	20.461
Crediti	363.297	0	363.297
Disponibilità liquide	288	0	288
Ratei e risconti attivi	3.258	0	3.258
Totale attivo	409.042	0	409.042
PASSIVO			
Patrimonio netto	23.222	0	23.222
Fondi per rischi e oneri	184	0	184
Trattam. fine rapporto	460	0	460
Debiti	385.161	0	385.161
Ratei e risconti passivi	15	0	15
Totale passivo	409.042	0	409.042

Il prospetto evidenzia gli investimenti realizzati nell'anno, le rimanenze di merci e la situazione creditoria e debitoria. In

Figure 5: Example of a partially recognized table being split up into two tables. Blank cells between the two tables are not classified as having been recognized.



Tavola

Effetti degli interventi sul conto economico delle Amministrazioni pubbliche (1)

(milioni di euro)

ENTRATE	
Maggiori entrate	**24.810**
Contrasto all'evasione e all'elusione fiscale	8.150
Studi di settore	*3.290*
Ampliamento di basi imponibili	*2.130*
Riscossione di tributi iscritti a ruolo	*1.200*
Altro	*1.530*
Trasferimento di parte del TFR all'INPS	5.940
Contributi sociali	4.380
Tasse automobilistiche	1.150
Patto di stabilità interno (imposte comunali)	1.110
Tassazione dei redditi finanziari (legge delega)	1.100
Aumento contributi per regolarizzazione immigrati	770
Disposizioni in materia di giochi	690
Modifiche detraibilità auto (netto sentenza CGE)	120
Sanità - effetto netto	110
Successioni e donazioni	60
Altre entrate tributarie	380
Altre entrate extratributarie	850
Minori entrate	[illegible]

SPESE	
Minori spese	**11.070**
Spese correnti	9.490
Patto di stabilità interno	*3.260*
Sanità	*2.950*
Consumi intermedi e trasferimenti dei Ministeri	*2.370*
Pubblico impiego	*390*
Altro	*520*
Spese in conto capitale	1.580
Ministeri	*830*
Altro	*750*
Maggiori spese	**14.680**
Spese correnti	8.040
Forze armate	*1.350*
Assegni familiari	*970*
Pubblico impiego - rinnovi contrattuali	*1.090*
Trasferimenti a imprese pubbliche	*1.100*
Sostegno al settore dell'autotrasporto	*280*
TFR (prestazioni INPS)	*430*
Prestazioni sociali	*430*

Figure 6: Example of two horizontally neighbouring tables (or *sub-tables*) merged together.

5 Ground truthing issues

The problems inherent in ground truthing tabular datasets are well known and have been described in detail in [5]. In this section, we describe the difficulties encountered in ground truthing the PDF-TREX dataset by the following examples:

- **table headings not properly aligned with the columns containing the data** (Fig. 7)**:** in this figure, several figures are misaligned with their headings. For example, it is not immediately clear whether the figure 118.011 belongs to the Valore iniziale column, or belongs to its own column. On closer examination, it appears that this figure was mistakenly right aligned. Similarly, the erroneous left alignment of the column heading Totale causes confusion;

- **tables being split by intermediate headings** (Fig. 8)**:** are these separate tables, or do these subtables all belong to one single table? If the table was not interrupted by paragraph text, it was interpreted as a single table, which also corresponds to the interpretation used by Ruffolo and Oro. However, in this case, the following problem arises:

- **spanning column headings in non-ruled tables** (Fig. 9)**:** here, one often cannot tell from the layout alone how many columns are spanned by the text. Although the text may only be two columns wide, it could be seen to logically span all data cells or even the entire width of the table. Even with domain-specific knowledge, this can present an ambiguous situation;

- **spanning row headings in non-ruled tables** (Fig. 10)**:** here, the layout of the table might suggest that the heading of a group of rows only belongs to the top row of the group. But logically, the heading applies to the row(s) underneath too. In the example, the year and months apply equally to the rows following them;

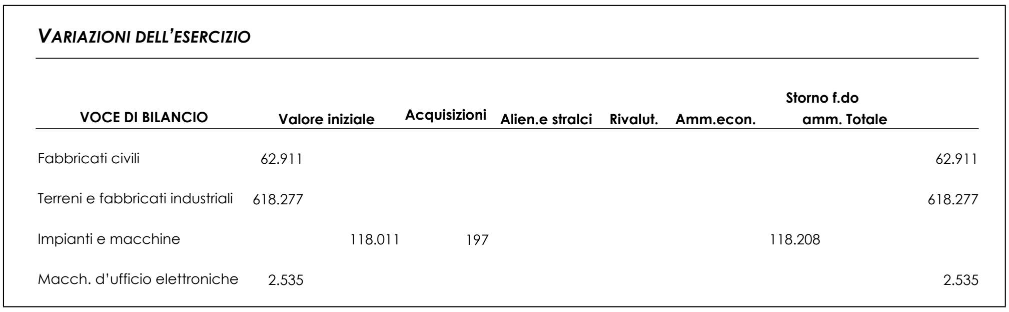
- **other tabulated data with leading dots** (Fig. 11)**:** in this example, the page contains two ruled tables and additional tabulated data inbetween these tables, which is presented using leading dots. This special type of formatting is usually reserved for special use-cases such as tables of contents and indices in books. Because the data presented is of a tabular nature, we did consider this to be a table in our ground truth, in contrast to Ruffolo and Oro. However, because this was one formatting convention we did not consider when designing our algorithm, this table was not detected at all by our system;

- **one line wrapped from previous table** (Fig. 12)**:** here, is appears that a single row (the "total" row) of a table on the previous page was wrapped over to the current page. Because we define tables as having a minimum dimension of 2×2, we did not class this "orphan row" as a table. This also corresponds to the decision made by Ruffolo and Oro.

We found that many of the above problems, such as misaligned columns and orphan rows, occurred due to poor, unprofessional typesetting of the documents in question. Some of these documents even proved troublesome for a human reader to understand, who could at least use his domain-specific knowledge to help the understanding process when the underlying logical structure cannot be determined from the layout alone (or, worse still, when the visual cues suggest a different logical structure to the correct one).

When we originally designed our system, we made two assumptions: Firstly, the input documents are correctly typeset and adhere to common layout conventions. Secondly, the logical structure of the data can be fully inferred from its layout. We found the PDF-TREX system to operate in a similar way, as it also encountered problems with the same documents.

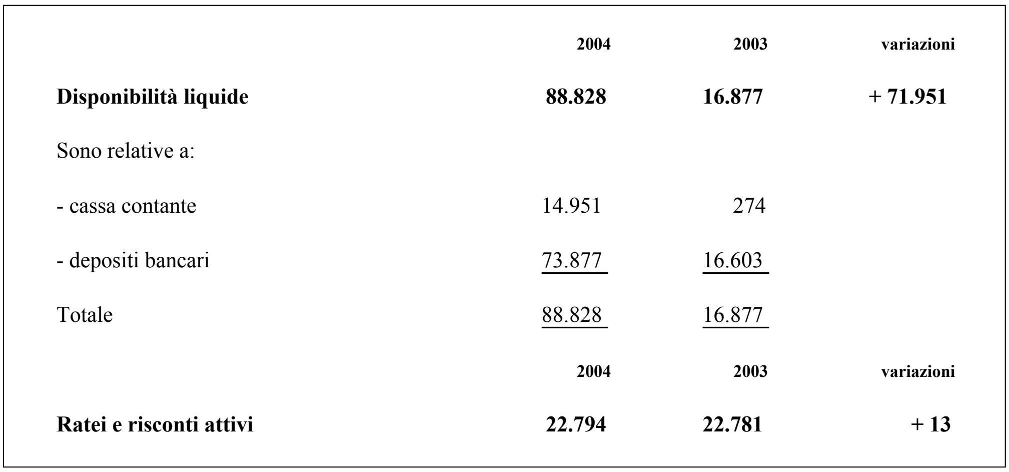
We therefore pose the following questions for consideration regarding the dataset:



VARIAZIONI DELL'ESERCIZIO

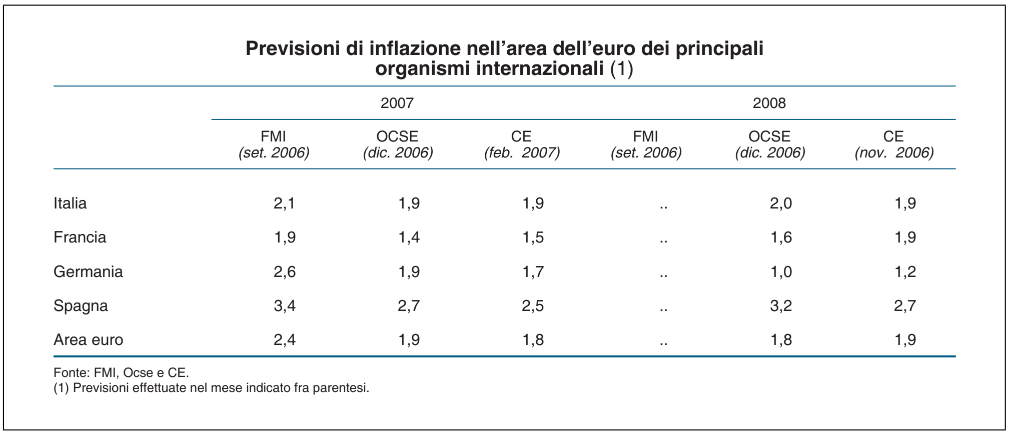
VOCE DI BILANCIO	Valore iniziale	Acquisizioni	Alien.e stralci	Rivalut.	Amm.econ.	Storno f.do amm.	Totale
Fabbricati civili	62.911						62.911
Terreni e fabbricati industriali	618.277						618.277
Impianti e macchine	118.011	197				118.208	
Macch. d'ufficio elettroniche	2.535						2.535

Figure 7: Example of a table with unclearly aligned columns



	2004	2003	variazioni
Disponibilità liquide	**88.828**	**16.877**	**+ 71.951**
Sono relative a:			
- cassa contante	14.951	274	
- depositi bancari	73.877	16.603	
Totale	88.828	16.877	
	2004	**2003**	**variazioni**
Ratei e risconti attivi	**22.794**	**22.781**	**+ 13**

Figure 8: Example of a table split by intermediate headings

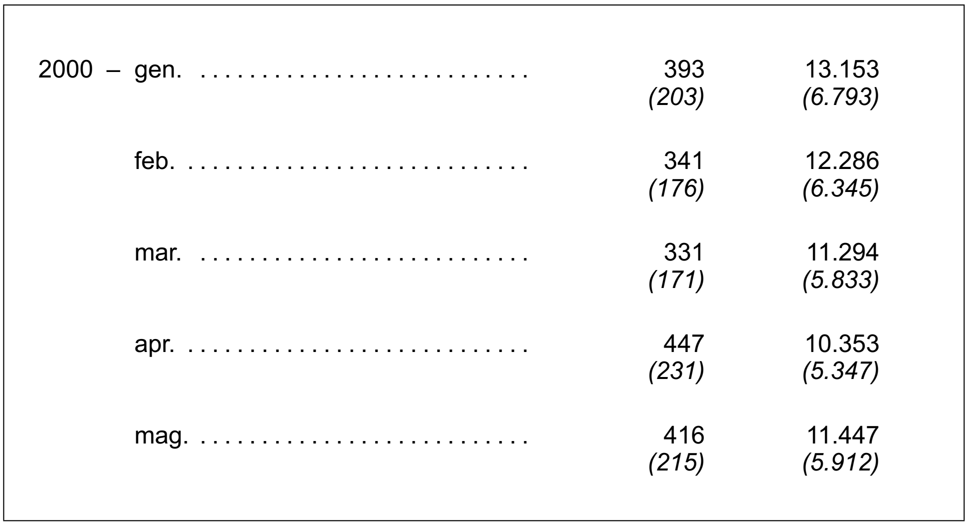


Previsioni di inflazione nell'area dell'euro dei principali organismi internazionali (1)

	2007			2008		
	FMI *(set. 2006)*	OCSE *(dic. 2006)*	CE *(feb. 2007)*	FMI *(set. 2006)*	OCSE *(dic. 2006)*	CE *(nov. 2006)*
Italia	2,1	1,9	1,9	..	2,0	1,9
Francia	1,9	1,4	1,5	..	1,6	1,9
Germania	2,6	1,9	1,7	..	1,0	1,2
Spagna	3,4	2,7	2,5	..	3,2	2,7
Area euro	2,4	1,9	1,8	..	1,8	1,9

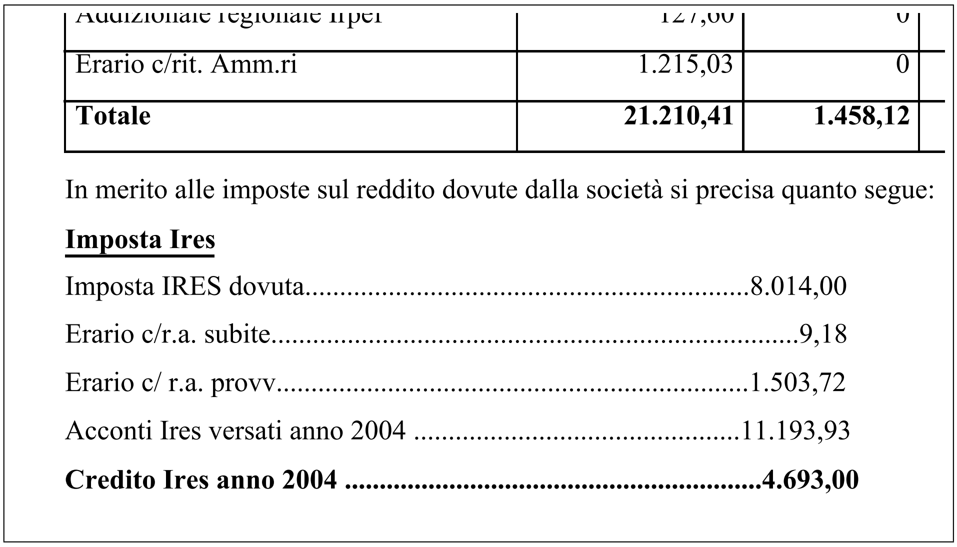
Fonte: FMI, Ocse e CE.
(1) Previsioni effettuate nel mese indicato fra parentesi.

Figure 9: Example of a table with spanning column headings



2000 – gen.	393 *(203)*	13.153 *(6.793)*
feb.	341 *(176)*	12.286 *(6.345)*
mar.	331 *(171)*	11.294 *(5.833)*
apr.	447 *(231)*	10.353 *(5.347)*
mag.	416 *(215)*	11.447 *(5.912)*

Figure 10: Example of spanning rows in an unruled table



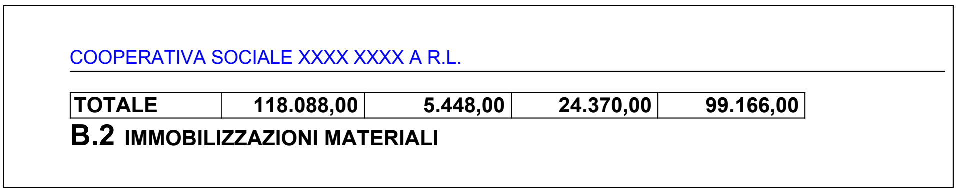
Addizionale regionale irpef	127,00	0
Erario c/rit. Amm.ri	1.215,03	0
Totale	**21.210,41**	**1.458,12**

In merito alle imposte sul reddito dovute dalla società si precisa quanto segue:

Imposta Ires

Imposta IRES dovuta...	8.014,00
Erario c/r.a. subite...	9,18
Erario c/ r.a. provv..	1.503,72
Acconti Ires versati anno 2004	11.193,93
Credito Ires anno 2004	**4.693,00**

Figure 11: Example of tabular data laid out using leading dots



COOPERATIVA SOCIALE XXXX XXXX A R.L.

TOTALE	**118.088,00**	**5.448,00**	**24.370,00**	**99.166,00**

B.2 IMMOBILIZZAZIONI MATERIALI

Figure 12: Top of a page containing a one-line table, presumably wrapped from the previous page

- Should documents containing obvious errors in their typesetting or layout (such as misaligned columns) be eliminated from the dataset?

- How do we deal with documents that have more than one correct interpretation of the ground truth?

- Perhaps a subset of documents could be defined, which are not reliant on domain-specific reasoning to be understood, and could be used to test systems which rely purely on the document's geometric structure. This could ensure less "noise" in the test results.

6 Aggregation of results

In common with a previous publication by Yildiz et al.[10], the results of the PDF-TREX system were given using separate precision and recall values for tables and cells. Here, the authors used a *document-based* approach: they first calculated the average table area and cell recall and precision for each document, and averaged these figures throughout the complete dataset. It is, however, not possible to calculate precision values for documents where no tables or cells were detected at all. We therefore decided to skip the step of calculating the averages for each document and create our totals by averaging the total numbers of detected cells directly over the complete dataset. With this method, documents containing more information (more table areas/cells) are also given more weighting in the final result.

A number of other approaches have also been proposed in the literature for aggregating the results of table structure recognition algorithms on different levels of granularity. Kieninger et al. [6, 7] propose a hierarchical model for representing the recognition result and the ground truth and redefine *table recall* and *table precision* based on their constituent objects. Thus, single values for recall and precision are returned. Because a strict hierarchy is used, only columns *or* rows can represented; the authors choose to represent columns as this better represents how their algorithm works.

A potential issue arises in the bottom level of the hierarchy, which is stated to be the *word level.* As the precise granularity or bottom-level segmentation may differ across different systems, difficulties could arise when comparing different systems to each other.

Hu et al. [4, 3] use a *directed acyclic graph* structure to represent the recognition result and ground truth. This structure can be used to represent both rows and columns simultaneously. Because of the complexity of the graph isomorphism problem, the two graphs are compared by a sequence of *random graph probing* operations, which need to be carefully defined according to the measuring criteria. Although such an approach is well suited for automatic tuning of algorithm parameters for a particular application, it is less suitable for comparing different systems to each other, not least because of the random element of this approach.

A further noteworthy approach is that of Cesarini et al. [1], who provide a formula for the *Table Location Index*, which combines correctly located, split and merged tables into a single score, and is used for automatic optimization of their algorithm. However, this approach does not deal with table cells, but only with table areas.

7 Discussion

The precision and recall measures including and excluding blank cells of both systems are shown in Table 3. The numerical results after reclassifying certain categories to better reflect the usability of the result are shown in Table 4.

Broadly speaking, the results show that whereas the PDF-TREX system achieves better cell recall, our system achieves better table area recognition results and better precision (i.e. fewer false positives) overall. The largest differences can be observed in the precision of table areas and the recall of table cells. After redefining certain cell classifications as false positives, a significant decrease was noticed in the numerical results of the PDF-TREX system. This is because the new definitions give a higher penalty to errors which would likely hinder data extraction. Excluding blank cells led to higher numerical results for precision and recall, and gave the PDF-TREX system a slight advantage.

It is worth mentioning that, during testing, it was clear that many documents which caused problems for our system also caused problems for PDF-TREX and vice versa, which suggests that both systems work in a similar way. It is believed that a small amount of fine-tuning of both algorithms, for example by adjusting thresholds or by trading off precision for recall, could lead to significantly better numerical results. The higher table cell recall of PDF-TREX could partly be attributed to the fact that several large tables were not detected at all by our system; these same tables were detected by PDF-TREX but split into several individual tables, which explains its significantly worse table area precision.

The significant drop in table area precision of PDF-TREX after reclassifying merged tables as false positives could be explained by one particular document in the dataset, which contained 12 tables on one page. Whereas

Table areas		**Our system**	**PDF-TREX**
Total areas		126	
Found correctly	TP	67	53
Data cells found	TP	22	17
Partially found	TP/FP	13	3
Split table	TP/FP	8	24
Extra table (from split)	FP	12	49
Incorrect table	FP	19	22
Merged into surroundings	TP	8	9
Merged	TP/FP	1	4
Not recognized	TN	9	16

Table cells		**Our system**	**PDF-TREX**
Total cells		10120 (9185 data; 935 blank)	
Found correctly data	TP	7489	8217
Found correctly blank	TP	718	806
Split full	TP	431	411
Split data	TP/FP	159	204
Split blank	TP	13	27
Extra data	FP	228	266
Extra blank	FP	935	1260
Incorrect data	FP	105	291
Incorrect blank	FP	83	534
Merged	TP/FP	28	104
Not recognized data	TN	1004	162
Not recognized blank	TN	202	83

Table 2: Totals of table area and cell classifications in the dataset

Table areas:

	Recall	**Precision**	***F*-meas.**
Our system	93.0%	78.8%	85.3%
PDF-TREX	87.5%	61.5%	72.3%

Table cells:

	Including blank cells			*Excluding blank cells*		
	Recall	**Precision**	***F*-meas.**	**Recall**	**Precision**	***F*-meas.**
Our system	87.3%	86.7%	87.0%	88.3%	96.1%	92.0%
PDF-TREX	96.5%	80.7%	87.9%	97.2%	94.2%	95.7%

Table 3: Precision and recall results of both systems for table areas and table cells

Table areas after reclassification:

	Recall	**Precision**	***F*-meas.**
Our system	76.9%	66.7%	71.4%
PDF-TREX	67.7%	47.6%	55.9%

Table cells after reclassification:

	Including blank cells			*Excluding blank cells*		
	Recall	**Precision**	***F*-meas.**	**Recall**	**Precision**	***F*-meas.**
Our system	87.3%	86.7%	87.0%	86.2%	96.0%	90.8%
PDF-TREX	93.5%	78.1%	85.1%	94.1%	91.0%	92.5%

Table 4: Results of both systems, after certain cell categories were recategorized

these tables were detected correctly by our system, they were erroneously merged together into one table by PDF-TREX. The fact that our results were not generated by averaging the results for each document, but were averaged directly over the complete dataset, means that this error was given a much larger weighting in the final result.

8 Conclusion

In this paper, we have addressed the significant issue of *evaluating* systems for table detection and structure recognition. We have also described the problems inherent in ground truthing of the dataset and the various ways that the results of individual tables and documents can be aggregated to generate a single set of figures for the complete dataset.

The use of precision and recall measures from the information retrieval field to model errors in table structure recognition, as in [8, 10, 6, 7], can lead to many ambiguities. It is hoped that the classification methodology that we have proposed in Section 4 will lead to a more consistent interpretation of these measures in the future, enabling the results of competing systems to be compared directly.

References

[1] F. Cesarini, S. Marinai, L. Sarti, and G. Soda. Trainable table location in document images. In *ICPR 2002: Proceedings of the 16th International Conference on Pattern Recognition*, volume 3, pages 236–240, 2002. 17

[2] T. Hassan and R. Baumgartner. Table recognition and understanding from PDF files. In *ICDAR 2007: Proceedings of the 9th International Conference on Document Analysis and Recognition*, volume 2, pages 1143–1147, 2007. 2

[3] J. Hu, R. Kashi, D. Lopresti, and G. Wilfong. Table structure recognition and its evaluation. In *Proceedings of Document Recognition and Retrieval VIII*, 2001. 16

[4] J. Hu, R. Kashi, D. Lopresti, and G. Wilfong. Evaluating the performance of table processing algorithms. *International Journal on Document Analysis and Recognition*, 4(3):140–153, March 2002. 16

[5] M. Hurst. Layout and language: Challenges for table understanding on the web. In *Proceedings of the First International Workshop on Web Document Analysis*, pages 27–30, 2001. 12

[6] T. Kieninger and A. Dengel. Applying the T-Recs table recognition system to the business letter domain. In *ICDAR 2001: Proceedings of the 6th International Conference on Document Analysis and Recognition*, pages 518–522, 2001. 3, 4, 16, 19

[7] T. Kieninger and A. Dengel. An approach towards benchmarking of table structure recognition results. In *ICDAR '05: Proceedings of the Eighth International Conference on Document Analysis and Recognition*, pages 1232–1236, 2005. 3, 16, 19

[8] M. Ruffolo and E. Oro. PDF-TREX: An approach for recognizing and extracting tables from PDF documents. In *ICDAR 2009: Proceedings of the 10th International Conference on Document Analysis and Recognition*, pages 906–910, 2009. 2, 3, 4, 19

[9] M. Ruffolo and E. Oro. PDF-TREX dataset. `http://staff.icar.cnr.it/ruffolo/files/PDF-TREX-Dataset.zip`, September 2009 (Web). 2

[10] B. Yildiz, K. Kaiser, and S. Miksch. pdf2table: A method to extract table information from PDF files. In *IICAI 2005: Proceedings of the 2nd Indian International Conference on Artificial Intelligence*, pages 1773–1785, 2005. 3, 16, 19