2011 International Conference on Document Analysis and Recognition

# A Method of Evaluating Table Segmentation Results Based on A Table Image Ground Truther

Yanhui Liang, Yizhou Wang
*National Engineering Lab. for Video Technology*
*Peking University*
*Beijing, China*
{*yhliang,Yizhou.Wang*}*@pku.edu.cn*

Eric Saund
*Palo Alto Research Center*
*Palo Alto*
*CA, USA*
*saund@parc.com*

***Abstract*—We propose a novel method to evaluate table segmentation results based on a table image ground truther. In the ground-truthing process, we first extract connected components from a given table image and connect them into an atom graph with weighed edges. Edge weight takes neighboring connected components' size similarities and distances into consideration. Then the ground truther semi-automatically determines the locations and spans of row/column separators according to projection profiles, under human supervision. We evaluate a given table segmentation by computing edit distance from its row and column separator assertions relative to ground truth. The edit distance is the sum of all the edit operation costs that correct wrong row and column separators. Each edit operation cost is a function of the sum of the weights of the edges that the separator cuts through. Thus, separator errors incur different costs depending on the severity of the error, where severity roughly corresponds to how forgivable the error would be considered by a human observer. Experimental results demonstrate that the proposed evaluation method is not only efficient, but also useful in formalizing the intuitive quality of different segmentations.**

***Keywords*-Table Segmentation; Evaluation; Edit Distance; Ground Truther;**

## I. Introduction

Tables play an important role in the representation, transfer and comparison of structured information in documents. With the recently growing sophistication of document analysis systems [1][2], more table processing algorithms have been introduced [3][4][5]. An efficient and accurate evaluation method is highly desired.

Several techniques for evaluating table processing results have been proposed recently [6][7]. Hu et al. [6] presented an evaluation system that represents tables, including both a processing result and ground truth, as directed acyclic attributed graphs. The system poses a series of queries and compares the responses for the two graphs. The essential limitation of this method is that attributed graph matching is difficult and error-prone and poses exponential worst-case running time. [8]. Also, the evaluation measure does not distinguish severity of errors since the evaluation criterion is expressed in terms of the number of correct answers for all probes.

As discussed by Embley et al. [9], more and more algorithms just aim at converting table images into editable Microsoft Excel or Word tables while concentrating little on tagging with logical labels to provide a semantic interpretation. We introduce an efficient evaluation method that focuses on the layout structure analysis of tables.

Our motivation is to provide a quantitative measure for a computed table segmentation, in comparison to groundtruth, that takes the severity (or forgivability) of errors into account.

Based on a ground truther we developed, the proposed method adopts an edit distance as the quantitative evaluation. In the ground-truthing process, we first extract atoms (connected components) from a given table image and connect them into an atom graph with weighed edges. The weight, which indicates a "binding force" between atoms, is computed by taking the atoms' size similarity and their spatial distance into consideration. Then, we find row and column separators semi-automatically according to horizontal and vertical projection profiles, and correct segmentation errors interactively. Each separator is assigned a weight which sums all the weights of the edges it cuts through. Next, we assess a given segmentation result by computing the edit distance from its row and column separator assertions to those of the ground truth. We identify three error types for separators in the segmentation result, namely, missing separator, spurious separator, and redundant separator. The cost function for an error-corrected operation considers the weight of the edited separator. A missing separator that is non-obvious due to its cutting through many high-weight edges is penalized less than a missing separator between sets of connected components that are not closely linked to one another. Spurious separators are more costly when they cut many highly weighted edges. Experimental results demonstrate that the proposed evaluation method is not only efficient, but also capable of providing intuitively satisfying accounts for the qualities of different segmentation proposals.

The remainder of this paper is organized as follows. We describe our evaluation method in detail in Section II. In

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.58

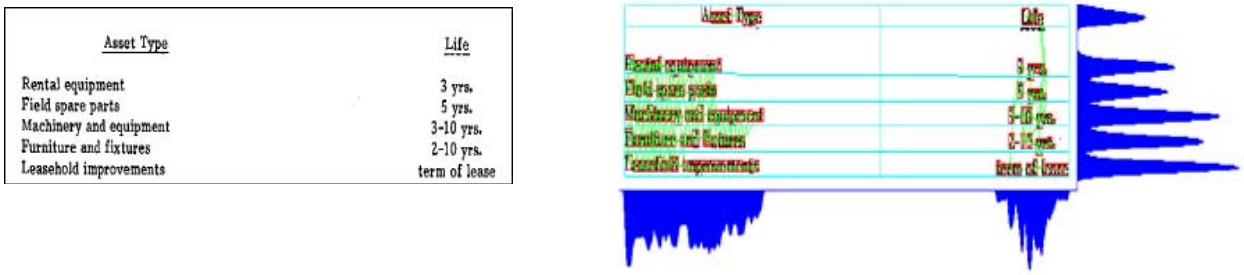



Figure 1. (a) Original table image. (b) Ground truth.

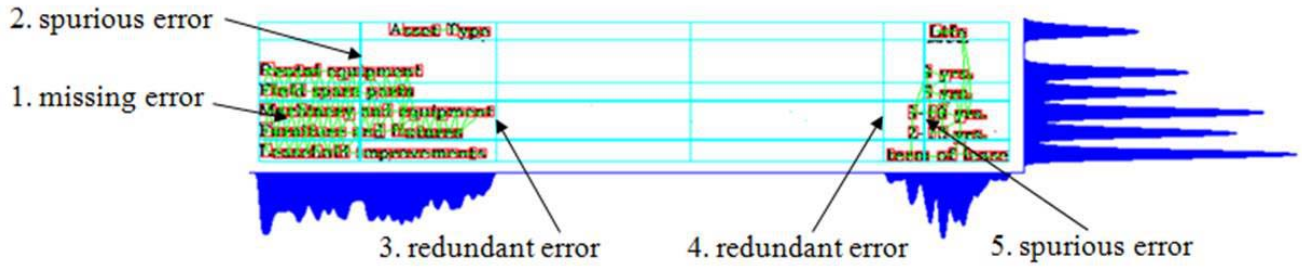



Figure 2. A segmentation result with various error types (errors are shown in format "id. error type"). The result is generated according to the vertical and horizontal projection profiles of the table.

Section III, a ground-truthing system is introduced as the basis for the evaluation algorithm. Section IV presents several examples of applying our evaluation method to candidate table segmentation results. Section V offers conclusions and discussion.

## II. Evaluation Method

In this section we present our approach to evaluating table segmentation results. Our method adopts an edit distance measure which identifies three error types appearing in the segmentation result and assigns a quantitative measure for each incorrect row/column separator. The method returns the weighted edit distance of the segmentation with respect to ground truth as the evaluation measure.

### A. Edit Distance Measure

We input a given table segmentation result in image form, with rule lines indicating separators. To do this, we extract connected components from the table image and judge whether a connected component is a row separator or not according to the width and the ratio of the height to the width of its bounding box. If its width is smaller than a threshold (3 pixels in our algorithm) and the ratio is bigger than a threshold (10 pixels we set), we recognize it as a row separator. We locate the column separators in a similar way by exchanging the width and the height.

Next we match the given segmentation result with ground truth by finding correspondences between the detected separators and the ground-truth separators. If the location and span of a detected separator matches completely with its corresponding ground-truth separator, we say they are in perfect correspondence. Otherwise, a segmentation error may have occurred. We classify the segmentation errors of separators into three categories: missing separator, spurious separator, and redundant separator. As depicted in Figure 2, a missing separator error (1. missing error) happens when no separator in the segmentation result matches its counterpart in the ground truth. A spurious separator (2. spurious error and 5. spurious error) is the one who makes a wrong cut of the table structure. A redundant separator (3. redundant error and 4. redundant error) is an extra separator that appears in one whitespace channel of the table image.

The edit distance $D$ between a segmentation result and ground truth is the sum of weighted penalties for different error correction operations.

$$D = \sum_{s_m \in M} c_m(G, s_m) + \sum_{s_s \in S} c_s(G, s_s) + \sum_{s_r \in R} c_r(G, s_r), \quad (1)$$

where $G$ is an atom graph, and $s_m$, $M$, $s_s$, $S$ and $s_r$, $R$ indicate a missing separator, the set of missing separators, a spurious separator, the set of spurious separators, a redundant separator, and the set of redundant separators, respectively. The cost functions are $c_m(G, s_m)$, the cost of

correcting a missing separator; $c_s(G, s_s)$, the cost of correcting a spurious separator; $c_r(G, s_r)$, the cost of correcting a redundant separator.

The different error correction costs in the edit distance reflect the obviousness of a hypothetical separator at different vertical and horizontal locations across the table, as reflected in the weights of edges in the atom graph cut by a separator.

$$c_m(G, s_m) = \frac{\omega_{max} - \omega_{s_m}}{\omega_{max}} \tag{2}$$

$$c_s(G, s_s) = \frac{\omega_{s_s}}{\omega_{max}} \tag{3}$$

$$c_r(G, s_r) = \frac{\omega_{max} - \omega_{s_r}}{\omega_{max}} \tag{4}$$

where $\omega_{s_m}$, $\omega_{s_s}$ and $\omega_{s_r}$ indicate the edge-cutting weights (see subsection II-B for details) of $s_m$,$s_s$ and $s_r$, respectively. For candidate row separators, $\omega_{max}$ is set as the maximum edge-cutting weight of row lines of the table image. For candidate column separators, $\omega_{max}$ is set to the maximum edge-cutting weight of column lines of the table image. We describe these four items in detail in the following subsections.

For tables with simple grid structure, we execute the algorithm and return the edit distance as the evaluation result. For tables with nested structure, we evaluate the segmentation in a coarse-to-fine recursive manner. We first detect row and column separators that cut through the entire table and compute the costs of the involved edit operations. Then we go to the "super-cells" which contains a smaller table structure, to continue the evaluation process. We repeat the above procedures until all super-cells are evaluated. We finally sum all the editing costs and return the edit distance.

Since the proposed method performs the evaluation by examining all separators occurring in both the given segmentation result and ground truth, the complexity of our algorithm is $O(n^2)$, where $n$ is the number of separators appearing in the segmentation result and the ground truth.

The counts and costs of segmentation errors of different types provide more information about the quality of the segmentation result and makes subsequent analysis of the table more convenient and efficient.

### *B. Separator Edge-cutting Weight*

To compute the edge-cutting weight of a row/column separator, we first detect the connected components (atoms) of the table image and build an atom graph out of them by applying a Voronoi-like algorithm [10]. Word-size atoms would perform equivalently in terms of edge weights, but in order to establish correct neighborhood relations, the triangulation would have to be among point samples at the atoms' perimeters instead of their centers.

The atom graph $G$ is denoted as: $G =< V, E >$.

The vertices of the atom graph are expressed as

$$V = \{a_i = ((x_i, y_i), h_i, w_i), i = 1, ..., N\} \tag{5}$$

in which $(x_i, y_i)$ is the centroid coordinate of the atom $a_i$; $h_i$ and $w_i$ are $a_i$'s bounding box height and width, respectively; $N$ is the number of the atoms in the table image.

The neighborhood structure is specified by the edge set

$$E = (e_{ij} : a_i, a_j \in V) \tag{6}$$

where $e_{ij}$ is the edge connecting atoms $a_i$ and $a_j$.

Each edge is assigned a weight, $w(e_{ij})$, which indicates the "binding force" between the pair of neighboring atoms. The weight is determined by the following factors:

1. The spatial distance of the pair of atoms, $\delta_{ij}$.
2. Table image projection profiles, $\pi$. If the bin height of the projection profile at the edge location is $\pi_{ij}$ and the global maximum bin height is $\pi_{max}$, the weight due to this factor can be defined as $\pi_{ij}/\pi_{max}$.
3. Size similarity of the pair of atoms, $\varepsilon_{ij}$, which is defined as $|h_i - h_j| \,/\, |w_i - w_j|$.

Thus, the edge weight can be expressed as

$$w(e_{ij}) = \lambda_0 \exp\{-\delta_{ij}\} + \lambda_1 \pi_{ij}/\pi_{max} + \lambda_2 \exp\{-\varepsilon_{ij}\} \tag{7}$$

$$\sum_k \lambda_k = 1, k = 0, 1, 2 \tag{8}$$

where $\lambda_k$ is the weight to balance the different cues. It can be simply set to equality.

The cutting-edge weight of a row/column separator is computed based on the weight of the edges it cuts through. It can be computed as:

$$\omega_s = \sum_{e_{ij} \in E} w(e_{ij}) \tag{9}$$

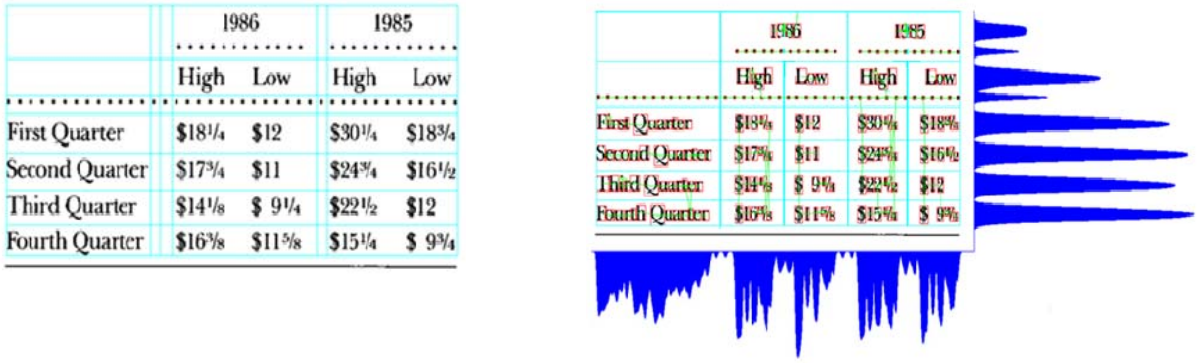
where $E$ is the set of weighed edges that separator $s$ cuts through.

Table I
EDGE-CUTTING WEIGHT FOR EACH UNCORRECT SEPARATOR (SHOWN IN FIGURE 2). HORIZONTAL $\omega_{max} = 5.414$ AND VERTICAL $\omega_{max} = 4.036$.

| Separator ID | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Edge-cutting weight | 4.539 | 3.231 | 0 | 0 | 2.914 |
| cost $c$ | 0.162 | 0.800 | 1 | 1 | 0.722 |

To find $\omega_{max}$, we use the horizontal and vertical projection profiles as search clues. We observe that the line with maximum edge-cutting weight is always located near the site of the maximum bin height of the projection profiles. So we search for the maximum edge-cutting weight for the row/column at the locations of the peak of horizontal/ vertical projection profiles, half character distance horizontally or vertically away from the peak.

Thus, we can calculate the edge-cutting weight for each error separator shown in example Figure 2 by setting each



| | 1986 | | 1985 | |
|---|---|---|---|---|
| | High | Low | High | Low |
| First Quarter | $18 1/4 | $12 | $30 1/4 | $18 3/4 |
| Second Quarter | $17 3/4 | $11 | $24 3/4 | $16 1/2 |
| Third Quarter | $14 1/8 | $ 9 1/4 | $22 1/2 | $12 |
| Fourth Quarter | $16 3/8 | $11 5/8 | $15 1/4 | $ 9 3/4 |

(a) The segmentation result.

(b) The ground truth.

Figure 3. Table segmentation result and its corresponding ground truth

$\lambda_k$ equally for every weighed edge that the separator cuts through. Table I shows the result.

Summing edit costs yields in Figure 2 yields a final edit cost of 3.684.

## III. Semi-automatic Ground Truther

To ground-truth the physical structure segmentation of tables, we have developed a semi-automatic ground truther. If we can detect rule-line row and column separators in the table image, we adopt them directly. Otherwise, the ground truther first determines the locations of candidate row/column separators automatically according to the horizontal and vertical projection profiles of table image. If any conspicuous segmentation errors happen, we correct them manually.

We adopt two stages to segment tables' physical structure. The first is a table image pre-processing step. In this stage, our ground truther first binarizes the table image, then applies a standard technique to extract connected components, and finally runs a Voronoi-like algorithm [10] to connect the extracted atoms in an atom graph.

The next stage is structure segmentation. The table is segmented automatically based on the results of its horizontal and vertical projection profiles. We define the locations of the row/column separators as the valleys of table image's horizontal-vertical projection profiles and compute the span of each separator by locating its starting and ending points. Then, conspicuous segmentation errors are corrected manually based on the interactive interface if necessary.

Figure 1 shows a table image and its physical structure segmentation result produced by our ground truther. In Figure 1(b), the small red rectangles around characters are the bounding boxes of the atoms and the connections between them shown in green are weighed edges of the atom graph. Projection profiles of this table image are shown diagrammatically as the dark blue graphs outside of the table. The light blue cut-lines locate the row and column separators of the table.

## IV. Experimental results

To demonstrate the effectiveness of this method for evaluating table segmentation results, we built a dataset containing 360 tables from various document images. An example is shown in Figure 3.

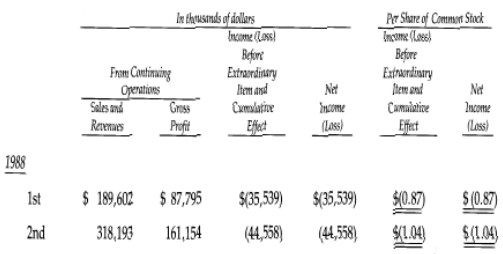
Figure 3(a) shows a segmentation result for a complex table which is generated by its horizontal and vertical projection profiles with a different threshold. We obtain its corresponding ground truth from our ground truther as shown in Figure 3(b). By executing the evaluation algorithm, one row separator and three column separators are found to be redundant and two column separators are missed. So the algorithm executes an editing sequence to transform the segmentation result into the ground truth. The maximum edge-cutting weight of the row is 3.736 and 2.978 for the column. By summing the weights of edges the edited separator cuts through, we calculate the edge-cutting weights for each edited separator, i.e. 0, 0, 0, 0.278, 0 and 0.334, respectively. The resulting edit distance is 7.794.

From the edit distance and the edge-cutting weight of each edited separator, we can see that the three redundant separators in the segmentation result cause erroneous splitting of the some columns. Missing separators reflect failure to distinguish other columns, which would lead to misinterpretation of the structure as well as the cell content of the table.

Figure 4 displays another table, its ground truth segmentation, and two example segmentations produced by hand. The first result appears more similar to ground truth than the second, and its edit distance is significantly smaller.

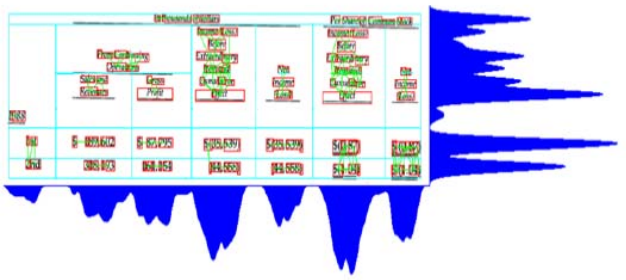
## V. Conclusion

This paper presents a method for evaluating table segmentation results based on a table image ground truther. The method distinguishes three segmentation errors types, and returns an edit distance reflecting the quality of different segmentation results. Visually conspicuous errors tend to

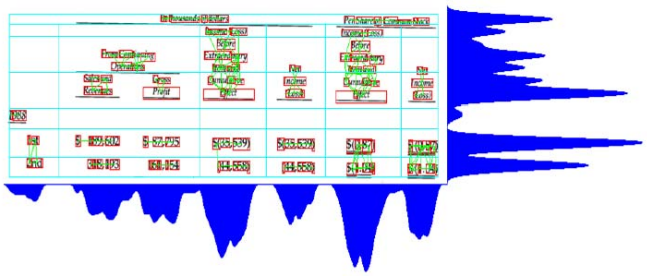


| | In thousands of dollars | | | | Per Share of Common Stock | |
|---|---|---|---|---|---|---|
| | From Continuing Operations | | Income (Loss) Before Extraordinary Item and Cumulative Effect | Net Income (Loss) | Income (Loss) Before Extraordinary Item and Cumulative Effect | Net Income (Loss) |
| | Sales and Revenues | Gross Profit | | | | |
| 1988 | | | | | | |
| 1st | $ 189,602 | $ 87,795 | $(35,539) | $(35,539) | $(0.87) | $ (0.87) |
| 2nd | 318,193 | 161,154 | (44,558) | (44,558) | $(1.04) | $ (1.04) |

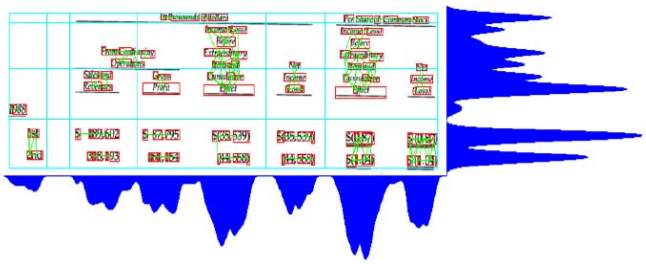
(a) Original table image.



(b) Ground truth.



(c) 1st parsing result with edit distance 3.812.



(d) 2nd parsing result with edit distance 8.473.

Figure 4. Comparison of the weighted edit distance for two sample segmentation results.

produce greater edit costs. We collected a dataset containing hundreds of table images to validate the performance of our evaluation method. Experimental results demonstrate its efficacy and accuracy.

A few issues deserve further discussion. The first is atom similarity. By employing OCR, we could incorporate more features beyond size similarity, to form a more general "cohesion" measure, including, for example font type, boldness, italics, digit vs. text.

A second issue is the recognition of redundant separators. If multiple separators are found in one wide whitespace channel of the table image, which one should be selected as the correct separator? Currently, we choose the one closest to its corresponding separator in the ground truth. But, if the valley of the projection profile is very wide, perhaps any separator in the channel may be considered equivalently correct.

These issues merit investigation in future work. In addition, a more rigorous study of the correlation of our evaluation measures in comparison to human perception is in order.

## REFERENCES


[1] K.Itonori. Table structure recognition based on textblock arrangement and ruled line position. Proc. Second Intl Conf. Document Analysis and Recognition, pp.765-768, Tsukuba Science City, Japan, 1993.

[2] W.Kornfeld and J. Wattecamps. Automatically locating, extracting and analyzing tabular data. Proc. Twenty-first Intl ACM SIGIR Conf. Research and Development in Information Retrieval, pp.347-348, Melbourne, Australia, 1998.

[3] D. Lopresti and G. Nagy. Automated table processing: An (opinionated) survey. In Proceedings of the Third IAPR International Workshop on Graphics Recognition, pp.109-134, Jaipur, India, September 1999.

[4] C. Peterman, C. H. Chang, and H. Alam. A system for table understanding. In Proceedings of the Symposium on Document Image Understanding Technology, pp. 55-62, Annapolis, MD, 1997.

[5] K.Zuyev. Table image segmentation. Proc. Fourth Intl Conf. Document Analysis and Recognition, pp.705-708, Ulm, Germany, 1997.

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

[7] F. Kboubi, A. H. Chabi, and M. B. Ahmed. Table recognition evaluation and combination method. In Proceedings of the 8th International Conference on Document Analysis and Recognition (ICDAR), pp. 1237C1241, Seoul, Korea, 2005.

[8] Steven S. Skiena. Geometric Probing. Science, Urbana, IL ,1988

[9] D.W. Embley, M. Hurst, D. Lopresti, and G. Nagy. Table processing paradigms: A research survey. International Journal of Document Analysis and Recognition, 8(2-3):66-86, June 2006.

[10] K. Kise, A. Sato, and M. Iwata, Segmentation of page images using the area Voronoi diagram, CVIU, 1998.