Semi-Automatic Ground Truth Generation For Chart Image Recognition

Li Yang[1], Weihua Huang[1] and Chew Lim Tan[1]

[1] School of Computing, National University of Singapore
3 Science Drive 2, Singapore 117543
Email: {yangli, huangwh, tancl@comp.nus.edu.sg}

Abstract. While research on scientific chart recognition is being carried out, there is no suitable standard that can be used to evaluate the overall performance of the chart recognition results. In this paper, a system for semi-automatic chart ground truth generation is introduced. Using the system, the user is able to extract multiple levels of ground truth data. The role of the user is to perform verification and correction and to input values where necessary. The system carries out automatic tasks such as text blocks detection and line detection etc. It can effectively reduce the time to generate ground truth data, comparing to full manual processing. We experimented the system using 115 images. The images and ground truth data generated are available to the public.

1 Introduction

In recent years, a number of research works have been done in the area of chart image recognition. Futrelle et al reported their work on diagram understanding [1] back in 1992. Yokokura et al also reported the work on x-y axes detection in chart images [2]. Zhou et al proposed methods chart type determination based on Hough transformation and learning-based approach [3, 4]. We also reported our own model based approach for chart type determination and understanding [5]. While the research activities are continuously carried out and new results are being reported, there does not exist a suitable standard to evaluate the results. In other words, there is no quantitative measurement of the result reported, and there is no public test set and ground truth data available for comparing the results obtained from different systems. Thus, it is desired to develop a system that can generate ground truth data for evaluating the performance of chart image recognition systems from many aspects.

The system proposed here is semi-automatic, which means the system does most of the job automatically while user interactions are required during the ground truth generation process. Since a typical chart image contains both text and graphics, we generate ground truth data for both kinds of information so that they can be used for evaluating both text recognition and graphics recognition. Furthermore, since the ultimate goal of chart image recognition is to understand the logic role of the chart components and to extract the data values carried by

the chart, we also decided to generate ground truth for chart components and data values.

Figure 1 illustrates the main modules in the proposed system. Pre-processing is performed to the input image first, including text/graphics separation, edge detection, vectorization and text grouping etc. Then ground truth data from text level and vector level are then generated in two different modules. In the next step, data from text level and vector level are combined to generate chart level ground truth. Dashed arrows in the figure indicate that user interactions are required in the modules. In the end, the ground truth data generated for one chart image are stored into an XML document.

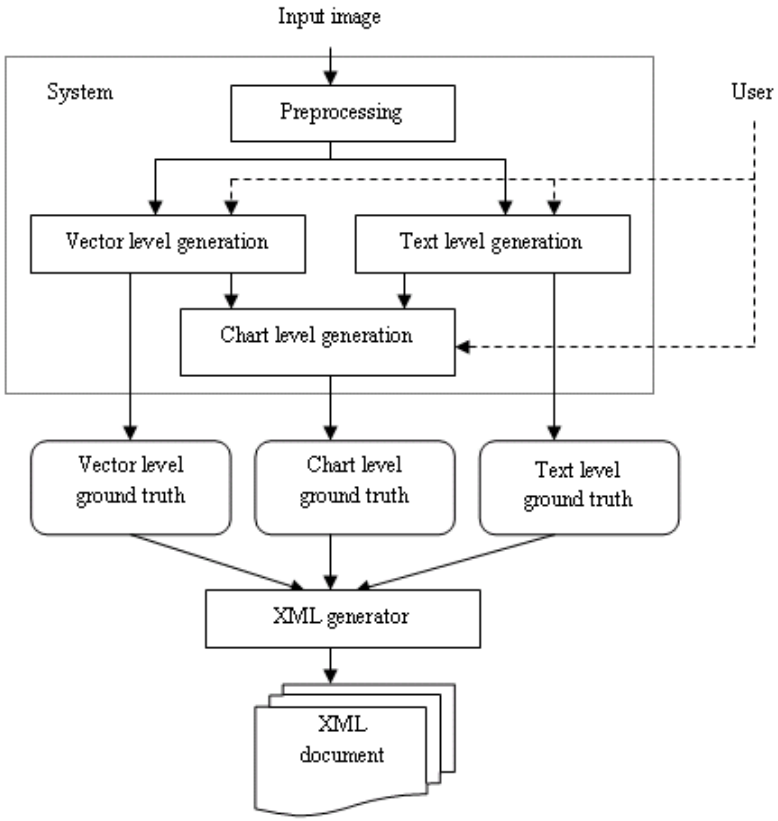

Fig. 1. The proposed ground truth generation system

The remaining sections of this paper will discuss the details of the proposed system. Section 2 summarizes some related works in ground truth generation. Section 3 introduces the detailed specification of chart ground truth data. Section 4 talks about how the ground truth data are generated. Section 5 presents the

results obtained together with some discussion. Section 6 talks about the issue of performance evaluation based on the ground truth data obtained. Section 7 concludes this paper with some future works mentioned.

2 Related Works

Ground truthing and performance evaluation has been recognized as an important factor in advancing the research in the document analysis field. In recent years, researchers proposed a number of systems for ground truth generation and performance evaluation for various kinds of documents. For example, Liu et al proposed a protocol for performance evaluation of line detection algorithms [6]. In the paper, they derived formulas for evaluating line detection accuracy on both pixel level and vector level. Wang et al introduced a system which automatically generates table ground truth and extracts table structure based on background analysis [7]. For evaluating the performance of text recognition systems, Zi and Doermann developed a system that generates document image ground truth from electronic text [8]. Yacoub et al presented their tool called "PerfectDoc" which is a ground truth environment designed for evaluating complex document analysis [9].

Depending on whether user interaction is required, ground truth generation systems can be divided into two categories: automatic and semi-automatic. The systems belonging to the former category perform all tasks in a fully automated manner thus are more efficient. Human correction can be carried out after the system generate a whole batch of ground truth data. However an requirement for the automatic approach is that the attributes involved in the ground truth should be either available at the beginning or easily obtainable. If this requirement can not be satisfied, then the semi-automatic approach should be adopted. In the case of chart images, some information are not available, such as the position and length of each line, and errors always exist when automatically obtaining such information. Thus our system belongs to the semi-automatic category.

3 Ground Truth Data in a Chart Image

As we mentioned previously, the ground truth data generated cover three levels: vector level, text level and chart level. The term "level" is used here to indicate different information granularity of the ground truth data. The order of granularity among various kinds of information in the chart image is:

$$\text{Pixel} < \text{vector} < \text{text} < \text{chart component}$$

Thus we define four levels of ground truth here. In the following subsections, we will discuss significance, essential attributes and availability of ground truth data at each level.

3.1 Pixel Level Ground Truth

Pixel level ground truth is useful especially for the evaluation of graphics recognition system. It can be used to evaluate the processing capability (Robustness) of image analysis algorithms [6]. The ground truth is basically the original clean image, and the actual image for testing is the degraded image. Since pixel level ground truth comes from clean original image, it may not always be available. For synthetic images, a clean image is available and the pixel values can be used as pixel level ground truth data. However, the images collected from web or scanned in already contain noise and distortions, thus the original pixel values become unknown. As the availability of ground truth for this level is not guaranteed, our system will not include it in the final output, though the image used for ground truth generation is still included.

3.2 Vector Level Ground Truth

Vector level ground truth is the line information in the images, or more precisely the attribute values of the straight line segments and arcs that form the lines. The essential attributes of straight line segments and arcs are the endpoints and the line width. With these attribute values, performance evaluation of vectorization algorithms can be achieved. Details about performance evaluation will be discussed in section 6. For both synthetic and real chart images, vector level ground truth data can be obtained. Although fully automatic extraction of line information is possible using existing vectorization algorithms, human effort is still needed here to manually correct the results to produce the final ground truth. Since higher level symbols (in our case the chart components) are often constructed from lines and arcs, vector level ground truth data not only serve as a standard to evaluate line detection algorithms, but also help to generate higher level ground truth data.

3.3 Text Level Ground Truth

We adopt the traditional representation of text level ground truth data, which consists of text zoning information and the electronic text content. A text zone is indicated by its four boundaries, and it is also an indication of the text location. In our system, each major text group is treated as a whole block and its bounding box is located. There are two reasons for doing so. Firstly, the human effort can be reduced by avoiding specifying the bounding boxes for each individual word. Secondly, it will be easier to assign logical role to a text block. Take the chart title as an example, a typical chart title may contain multiple words and the whole group of words has only one logical role in the chart image. Of course if the text zones are to be used to measure the segmentation capability of OCR systems, then one of the obvious adjustment here is to apply an automatic text segmentation algorithm (such as the x-y cut algorithm) to further locate the bounding box for each word in a text block.

3.4 Chart Level Ground Truth

A chart image has various components and features, but only a subset of them are essential for understanding the chart image. They are summarized in Figure 2. The title of a chart is not always available. If it is available, then it provides contextual information about the chart, together with other textual information in the chart. The axes only exist for some chart types, such as bar chart or line chart etc. Besides the position of each axis, axis title, labels along the axis and the axis range are also important for capturing complete axis information. If there are more than one data series presented in the chart, then the legend information is used to distinguish among data series. Legend information includes legend name and legend indicator. Data segments represent data value in different forms for different chart types. For example, there are bars in bar chart, pies in pie chart etc. So in the ground truth data, we not only present the name and value of each data segment, but also specify its form. In case there are more than one data series, the category each data segment belongs to is also recorded.

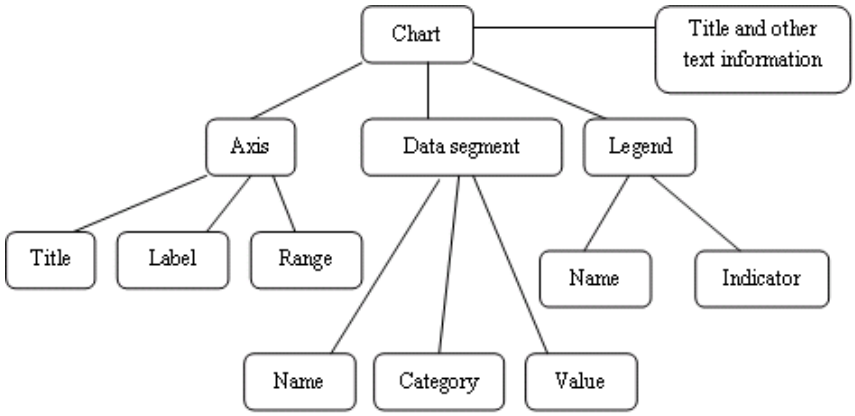

Fig. 2. Essential components in a chart image

4 Ground Truth Data Generation

4.1 System Preprocessing

There are several steps in the preprocessing stage:

1. Text/graphics separation. Textual information and graphical information are separated in this step using connected component filtering. A series of thresholds are applied to differentiate text components from graphical components. Most of the text components can be separated from graphics successfully.

Characters touching graphics cannot be separated in this case, but the problem can be partially solved later by finding user specified text regions. The text components are binarized and stored in a separate text image, which will be passed as input to text blocks construction step. The graphical components are kept in the original image and will be further analyzed to find the line information.

2. Edge detection. Since all vectorization methods require binary image, edge map is constructed in this step. To effectively identify the edges, the system needs to be given the maximum allowed edge thickness. Edge detection is done by calculating intensity differential among neighboring pixels, followed by gap filling between left edge and right edge.
3. Text blocks construction. Text block construction is based on the method described in [10] to the text components found previously. The system automatically calculates the text block candidates, after which the user can then refine the result by deciding whether to further split a block or merge some blocks.
4. Vectorization. The purpose of vectorization is to detect line information, or more precisely information of the straight lines and arcs. Here we use the vectorization methods proposed by Liu et al [11, 12] to construct straight lines and arcs respectively. The results are stored in the vector form. The vector of straight line contains starting point, ending point and line width. The content of the vector of arc is similar, except that the arc centre is also stored.
5. Locating Feature points. The feature points include endpoints of straight line segments and arcs, touching points of two lines, cross points and corner points, as illustrated in Figure 3(a) to (d). The point sets are calculated by the system automatically, and will be used as a basis for adjusting the user specified points. If there is more than one feature point near the user selected location, then the nearest feature point will be chosen as the final point.

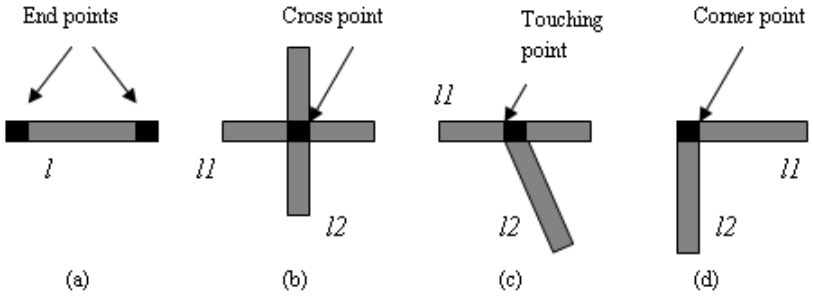

Fig. 3. (a)-(d). Illustration of the set of feature points

4.2 Vector Level Ground Truth Generation

As the vectors of straight lines and arcs are already available, the task of the user is to verify the correctness and accuracy of the vectorization result. The vectors are drawn on the original image and the user can manually adjust the endpoints of a vector if it is too long, too short or outside the original line. After the user verify and correct all the vectors, the information stored in the vectors is then saved as the vector level ground truth data. Furthermore, the vector information is also passed on for chart level ground truth generation.

4.3 Text Level Ground Truth Generation

User adjustment can be performed similarly to the text block candidates automatically identified by the system, by refining the boundaries of each candidate. If the text block candidate contains multiple text blocks, the user can manually specify the cutting point to separate them. On the other hand, if several text block candidates belong to the same text block, the user can also group them and form a larger block. For electronic text, current system relies on manual input to guarantee the correctness of the text content. The main reason is that currently we haven't built an OCR module in our system. However, this is not an expensive approach since usually the amount of text in a chart image is not large. To further improve the efficiency of our system, we can add the OCR module into the system, as part of the future work.

After all text blocks are fixed and the electronic contents are input, the information is saved as text level ground truth. The information will also be used when chart level ground truth is generated.

4.4 Chart Level Ground Truth Generation

As we mentioned, chart level ground truth contains the information of a set of essential chart components. Obtaining such information is not straightforward. The system has to rely on heuristic rules and user interactions to identify the exact position and attributes of the chart components and obtain their values.

To find the graphical chart components, the user just needs to indicate the rough position of the feature points for each component, and then the system will automatically find the precise position by finding the best feature point within a predefined range. If the feature point selected by the system is wrong, the user can still manually adjust the position of the point in four directions.

To find the textual chart components, the user needs to manually specify the correspondence between a text block and its logical role. It is difficult to automate this step, because the text/graphics correspondence is still being studied and no general solution is found yet. To obtain the data values, one way is to generate chart images based on synthetic data. In this way, the original data values are available for comparison with the extracted data values. For scanned chart images, the original data values may not be available, thus them need to be calculated based on the information available in the images. There are two

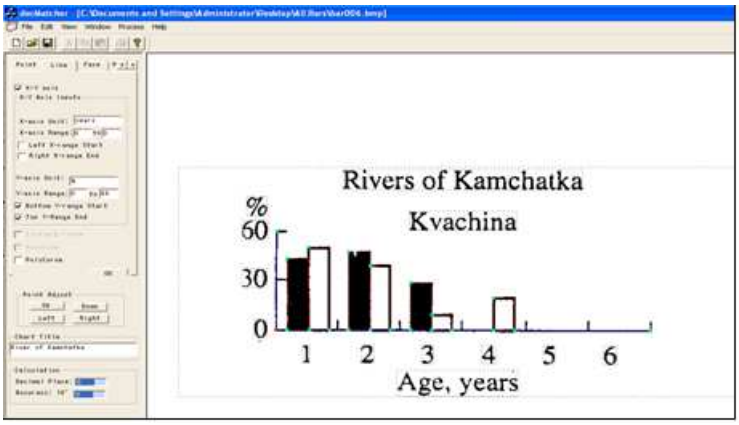

Fig. 4. Snapshot of the system interface

cases: if the data values exist in the image in the format of text objects, then the user can help to specify them by input to the system. If the data values are not directly given, then the system will calculate them and the user needs to verify and make correction if there is a large error.

Figure 4. shows a snapshot of the system interface. The input image is placed at the center of the image panel, surrounded by dash boundaries. The tool panels are on the left of the window. The green dots in the input image indicate the feature points used to specify the x-y axis and the bar components. The red dot is the origin of the coordinate system. The snapshot also shows the bounding boxes of all text blocks. The content and logic role of each text block can be specified in the tool panel.

5 Experimental Results and Discussion

5.1 The Data Set

Current data set contains 115 chart images. 75 of them were scanned chart images and the remaining 40 images were downloaded from the internet. The images are either greyscale or color images. The effect of noise results in blurred edge, extra dots and color distortion. More details are shown in Table 1.

5.2 Ground Truth Generated

The proposed system is applied to each image in the data set to generate its ground truth data. Table 2. shows some statistics of the ground truth generated. To generate XML format outputs, a set of tags were defined based on the elements in the ground truth data. The details about XML tag definition can be found in the XML files contained in the ground truth data released on the web, thus they are not discussed here due to limited space in this paper.

Table 1. The data set

	Scanned		Downloaded		
Chart type	Greyscale	Color	Greyscale	Color	**Total**
Bar chart	61	-	2	12	**75**
Pie chart	-	-	7	18	**25**
Line chart	14	-	-	1	**15**
Total	**75**	**0**	**9**	**31**	**115**

Table 2. Some statistics about the ground truth data generated

Granularity	**Elements**	**Total**
Vector level	Straight lines	4377
	Arcs	41
Text level	Text blocks	1587
	Words	2095
Chart level	Chart titles	58
	Axes	180
	Axis labels	1181
	Bars	840
	Pies	116
	Data points (for line chart)	131

5.3 Discussions

Since the original input image is noisy, the lines in the image have distorted edges, which may cause trouble for finding the correct line width. In the vectorization step, the system calculates the width of a line by taking the average width of all small segments in the line. To guarantee the accuracy of the line width detected, the user needs to manually verify it and adjust the line width to a most suitable value if necessary.

There are some special characters in the XML specification that cannot be displayed properly, such as "&" and "<" and ">". Thus to guarantee the completeness of information in the text level ground truth data, we changed these special characters to "and", "less_than" and "greater_than". But then the character set does not match the original set, so we also include a plain text version of the ground truth so that all characters are available, including the special characters.

One of our assumptions is that the lines in the chart components are solid lines. Although in most cases the assumption is valid, there are some exceptions. For example, dash line may be used to connect the data points in a line chart. And sometimes the axes also appear as dash lines. Thus to overcome this weakness, a dash line detection algorithm should be implemented and added to the vectorization process.

On average it took around two to three minutes to process an input image. It may seem a bit slow, comparing to fully automatic processing. But we should

consider the necessity of human interaction and correction involved. And the time consumed is definitely much shorter than complete manual processing.

6 Issues on Performance Measure

As the ground truth data become available, we also discuss here how the data can be used to measure the performance of an image recognition system. The system to be evaluated does not need to perform all the tasks and generate all the data to match with the ground truth. It can be a line detection system, a text recognition system or an image understanding system.

At the vector level, we refer to Liu et al's paper about performance evaluation of line detection algorithms [6]. According to Liu, line detection accuracy is indicated by the vector recovery index VRI, which can be obtained by calculating the line detection rate Dv and the false alarm factor Fv:

$$D_v = \frac{\Sigma_{g \in V_g} Q_v(g)\, l(g)}{\Sigma_{k \in V_d} l(g)} \tag{1}$$

where $Q_v(g)$ is total vector detection quality of ground truth vector g and l(g) is the length of the vector, V_g is the set of vectors in the ground truth and V_d is the set of vectors detected.

$$F_v = \frac{\Sigma_{k \in V_d} F_v(k)\, l(k)}{\Sigma_{k \in V_d} l(k)} \tag{2}$$

where $F_v(k)$ is the false alarm factor of the detected line k.

Thus the combined vector recovery index is defined as:

$$VRI = \beta D_v + (1-\beta)(1-F_v) \tag{3}$$

where β is the relative importance of detection and 1-β is the relative importance of the false alarm. More details on the term definitions and the formulas can be found in the original paper.

At the chart level, the detection rate of graphical data components can be obtained similarly by calculating the data component recovery index:

$$DRI = \mu D_d + (1-\mu)(1-F_d) \tag{4}$$

where μ is the relative importance of detection and 1-μ is the relative importance of the false alarm. And here:

$$D_d = \frac{\Sigma_{k \in C_g} D_d(k)\, S(k)}{\Sigma_{k \in C_g} S(k)} \tag{5}$$

where D_d is the overall detection rate, $D_d(k)$ is the detection rate for ground truth component k and S(k) is the size of ground truth component k, C_g is the set of graphical data components in the ground truth.

$$F_d = \frac{\Sigma_{k \in C_d} F_d(k)\, S(k)}{\Sigma_{k \in C_d} S(k)} \tag{6}$$

where F_d is the overall false alarm rate, $F_d(k)$ is the false alarm rate of the detected component k, C_d is the set of graphical data components detected. $D_d(k)$ and $F_d(k)$ are defined as:

$$D_d(k) = \frac{S(C_d(k) \cap C_g(k))}{S(C_g(k))} \tag{7}$$

$$F_d(k) = 1 - \frac{S(C_d(k) \cap C_g(k))}{S(C_d(k))} \tag{8}$$

where $C_d(k)$ is the detected component and $C_g(k)$ is the ground truth component.

For evaluation of text recognition results, well known IR metrics precision P and recall R are used instead of detection rate and false alarm. Calculation of the precision and recall for character recognition is straightforward:

$$P = \frac{|Ch_g \cap Ch_d|}{|Ch_d|} \tag{9}$$

$$R = \frac{|Ch_g \cap Ch_d|}{|Ch_g|} \tag{10}$$

where Ch_g is the set of characters in the ground truth text and Ch_d is the set of characters recognized. To evaluate the accuracy of text blocks detected, a slight change need to made to equation (9) and (10). Instead of the intersection between two sets, the overlap between two corresponding bounding boxes should be calculated.

The overall performance score S is defined as:

$$S = \Sigma_{i=1}^{n} w_i S_i \tag{11}$$

where Si is the individual score at a single level i, and w_i is the weight assigned to each $S_i(\Sigma w_i = 1)$. Equation (11) is still applicable for systems focusing on only one task, by turn off other performance measures (setting all other weights to zero).

7 Conclusion and Future Work

In this paper, we described our work of ground truth generation from scientific chart images. The system is semi-automatic, which requires user interaction to provide guidance and necessary input to the system and the system automatically does underlying calculation and refinement. The ground truth data can be used to evaluate the performance of document recognition system for various purposes, such as text recognition, graphics recognition and image understanding systems. Currently we have generated ground truth data for 115 images scanned in or downloaded from the internet. The amount will keep increasing as we conduct more and more testing in the future. The generated ground truth data is publicly available, through URL:

http://www.comp.nus.edu.sg/~huangwh/ChartRecognition/GroundTruth/

At the moment, OCR is not integrated into the system, so the content of each text box requires manual input. In the future, an OCR module can be included and then text can be automatically recognized. Then only manual correction is needed, which further minimize the human effort. Another improvement to be made is to include a text segmentation algorithm to automatically divide the text block into small boxes for each individual word.

References

1. R. P. Futrelle, I. A. Kakadiaris, J. Alexander, C. M. Carriero, N. Nikolakis, J. M. Futrelle: Understanding diagrams in technical documents, IEEE Computer, Vol.25, pp75-78, 1992.
2. N. Yokokura and T. Watanabe: Layout-Based Approach for extracting constructive elements of bar-charts, Graphics recognition: algorithms and systems, GREC'97, pp163-174.
3. Y. P. Zhou and C. L. Tan: Hough technique for bar charts detection and recognition in document images, International Conference on Image Processing, ICIP 2000, page 494-497, 2000.
4. Y. P. Zhou and C. L. Tan: Learning-based scientific chart recognition, 4th IAPR International Workshop on Graphics Recognition, GREC2001, page 482-492, 2001.
5. W. H. Huang, C. L. Tan and W. K. Leow: Model based chart image recognition, International Workshop on Graphics Recognition, GREC2003, 30-31 July 2003, Barcelona, Spain.
6. W. Liu, D. Dori: A protocol for performance evaluation of line detection algorithms, Machine Vision and Applications, 1997, vol. 9, pg. 240-250.
7. Y. Wang, R. M. Haralick, I. T. Phillips: Automatic Table Ground Truth Generation and a Background-Analysis-Based Table Structure Extraction Method. ICDAR 2001: 528-532
8. G. Zi, D. Doermann: Document Image Ground Truth Generation from Electronic Text, 17th International Conference on Pattern Recognition, ICPR'04, vol. 2, pp. 663-666.
9. S. Yacoub, V. Saxena and S. Sami: PerfectDoc: A Ground Truthing Environment for Complex Documents, 8th International Conference on Document Analysis and Recognition, ICDAR'05, vol. 1, pg. 452-456.
10. B. Yuan and C. L. Tan: A Multi-level Component Grouping Algorithm and Its Applications, 8th International Conference on Document Analysis and Recognition, ICDAR'05, pg. 1178-1181.
11. W. Liu and D. Dori: Sparse Pixel Vectorization: An Algorithm and Its Performance Evaluation", IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, vol. 21, pg. 202-215.
12. D. Dori and W. Liu: Incremental Arc Segmentation Algorithm and Its Evaluation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, vol. 20, pg. 424-431.