GREC 2007 Arc Segmentation Contest: Evaluation of Four Participating Algorithms

Faisal Shafait[1], Daniel Keysers[1], and Thomas M. Breuel[2]

[1] Image Understanding and Pattern Recognition (IUPR) research group
German Research Center for Artificial Intelligence (DFKI) GmbH
D-67663 Kaiserslautern, Germany
`faisal@iupr.dfki.de, keysers@iupr.dfki.de`

[2] Department of Computer Science, Technical University of Kaiserslautern
D-67663 Kaiserslautern, Germany
`tmb@informatik.uni-kl.de`

Abstract. Automatic conversion of line drawings from paper to electronic form requires the recognition of geometric primitives like lines, arcs, circles etc. in scanned documents. Many algorithms have been proposed over the years to extract lines and arcs from document images. To compare different state-of-the-art systems, an arc segmentation contest was held in the seventh IAPR International Workshop on Graphics Recognition - GREC 2007. Four methods participated in the contest, three of which were commercial systems and one was a research algorithm. This paper presents the results of the contest by giving an overview of the dataset used in the contest, evaluation methodology, participating methods and the segmentation accuracy achieved by the participating methods.

Keywords: Graphics Recognition, Line Drawings, Technical Drawings, Arc Segmentation Contest.

1 Introduction

Reliable detection of geometric primitives like lines, arcs, circles etc. in document images is one of the key problems in graphics recognition. Due to the importance of this task, the International Association for Pattern Recognition's Technical Committee on Graphics Recognition (IAPR TC10) has been organizing biennial arc segmentation contests since 2001 [1,2,3]. The purpose of these contests was to provide a platform for comparative evaluation of state-of-the-art research and commercial graphics recognition algorithms. The benchmarking of algorithms in this way helps in objectively evaluating the performance of participating systems and highlights the strengths and weaknesses of these systems. Therefore the contest-based approach for comparing algorithms is also used in other domains of document analysis research, like page segmentation [4], handwriting recognition [5], and document image dewarping [6].

This contest is fourth in the series of arc segmentation contests and was held at the seventh IAPR International Workshop on Graphics Recognition (GREC

2007), in Curitiba, Brazil, September 20-21, 2007. A dataset of five training and five test images was used in the contest. This contest was different from the previous three contests from the view point of ground-truth representation and performance evaluation protocol. Previous arc segmentation contests used the VEC format for representing arcs, and used the VRI score [7] as a measure of performance of arc segmentation. In this contest, we have used a color-based representation and evaluation scheme [8] discussed in Section 2 and Section 3. Four methods participated in the contest. A brief description of the methods is given in Section 4. The dataset used in the contest and the results of the contest are given in Section 5 followed by a conclusion in Section 6.

2 Representation of Geometric Primitives

The traditional way of representing geometric primitives like lines, arcs, or circles in a drawing is to use their parametric representation. The VEC format uses this representation in plain text form, using one line of parameters for each arc. Other drawing tools can then read this format and reproduce an image containing exactly the arcs given in the VEC-format text file. However, if some of the arcs are incorrect, it is hard to find the source of error, since the correspondence of the arcs in VEC-format to pixels in the original image can not be easily established. In addition, the performance of the algorithm can not be judged by looking only at the VEC-format text file, and specialized software is needed to view the detected arcs and analyze the segmentation errors.

To overcome these problems, we propose a new representation of arc segmentation. This representation is based on pixel-accurate color-coding of page segmentation as proposed in [8]. Arc segments in an image are represented within the image such that each pixel belonging to an arc is assigned as its value the index of the arc. A particular color can be assigned to the page background (e.g. `0xffffff`) and to all pixels not belonging to any arc (e.g. `0x000000`). This representation of arc segmentation is particularly convenient because it can be used to accurately represent different arcs in the same image as shown in Figure 1. Secondly, it can be saved and exchanged using any lossless color image format, thereby avoiding the need for specialized software for viewing the arcs. The assignment of colors to arcs is arbitrary, so any colors can be chosen for representing different arcs. Pixels belonging to more than one arc can be assigned a unique color if needed.

3 Vectorial Score for Performance Evaluation

To evaluate the performance of an arc segmentation algorithm, we use the vectorial score proposed in [8]. According to this vectorial score, different errors that are measured are:

Oversegmented arcs: the number of arcs that are either split into more than one arc, or are partially detected.

Undersegmented arcs: the number of arcs merged with some other arc.

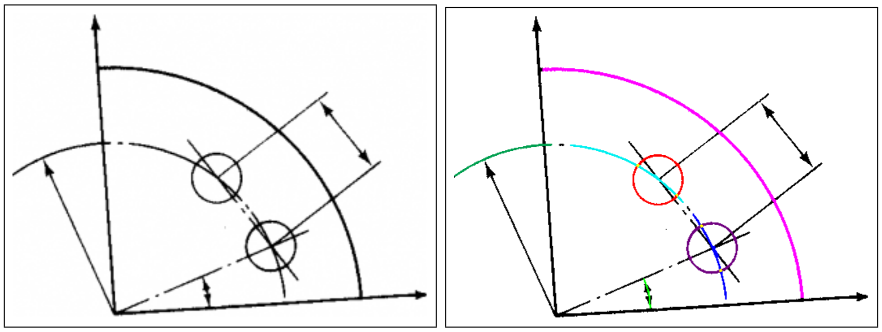


Fig. 1. An example image to demonstrate color encoding of arc segments. Each arc found in the image is labeled with a unique color.

Total Oversegmentations: the total number of segmentations that ground-truth arcs were split into.
Total Undersegmentations: the total number of segmentations that would be needed to split all merged arcs.
Missed arcs: the number of arcs that were not found by the algorithm.
False alarms: the number of detecting arcs originating from noise or non-graphics elements.

4 Participating Methods

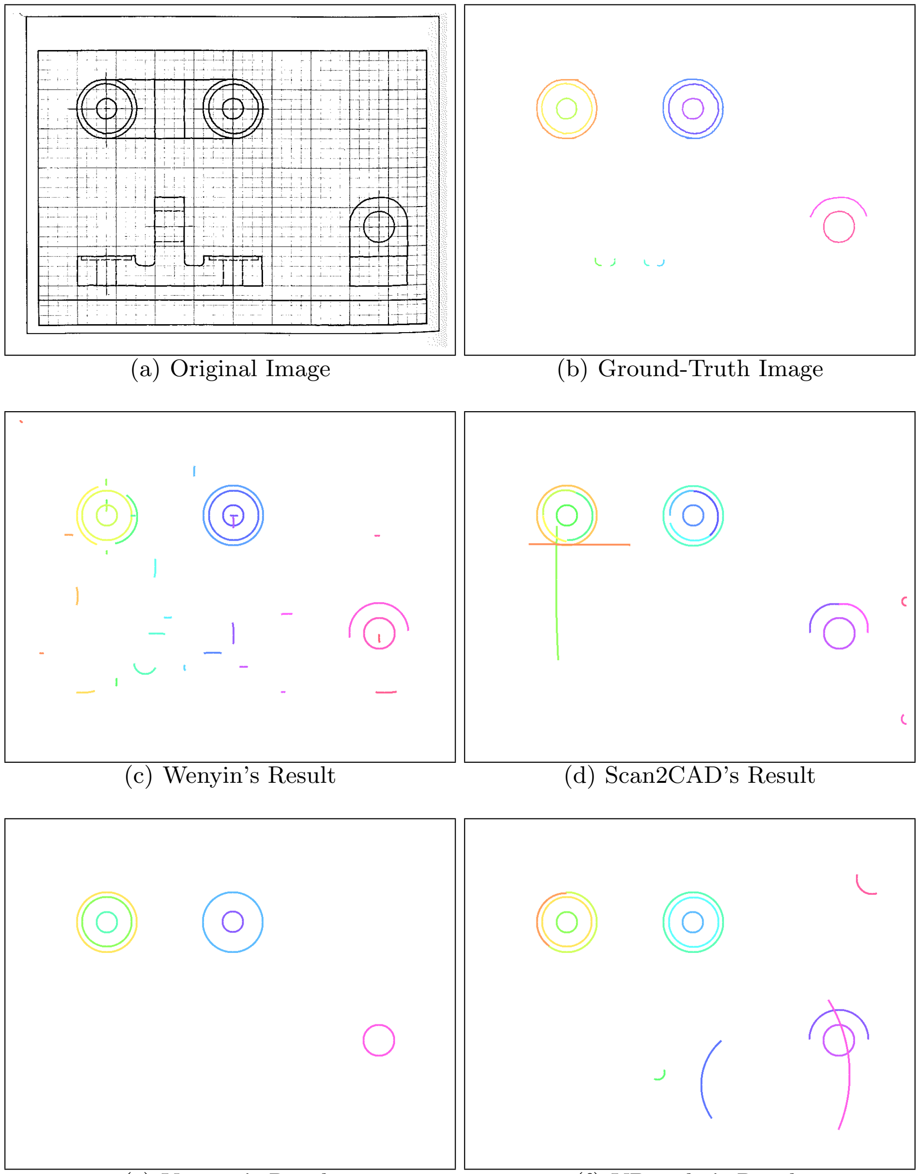
Results of four methods were presented for participation in the contest:

1. Liu Wenyin's method [9]
2. Vectory software ver. 5.0 (http://www.graphikon.de)
3. Scan2CAD software ver. 7.5d (http://www.softcover.com)
4. VPstudio software ver. 8.02 (http://www.softelec.com)

Liu Wenyin from City University of Hong Kong provided the results of his method, whereas Hasan Al-Khaffaf from Universiti Sains Malaysia presented the results of the other three commercial systems.

5 Results

The results of all participating methods on each test image are shown in Figures 2 to 6. The test images were obtained by scanning selected engineering drawings from different books. The ground-truth was then generated manually by coloring all the pixels belonging to an arc with a unique color using an off-the-shelf image manipulation program.

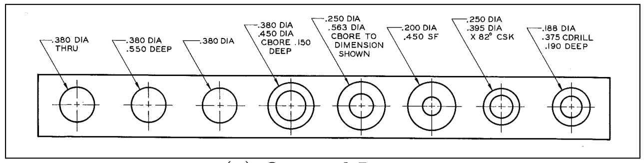


(a) Original Image (b) Ground-Truth Image

(c) Wenyin's Result (d) Scan2CAD's Result

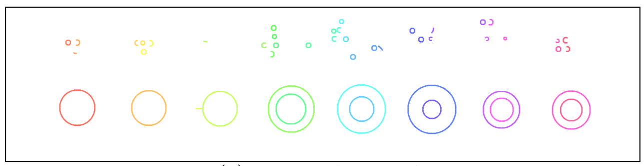
(e) Vectory's Result (f) VPstudio's Result

Fig. 2. Results of the four participating methods on the first test image. For clarity in the ground-truth image, only those foreground pixels that belong to an arc are shown. Due to the presence of the background grid, none of the participating methods could correctly segment all arcs from the image.

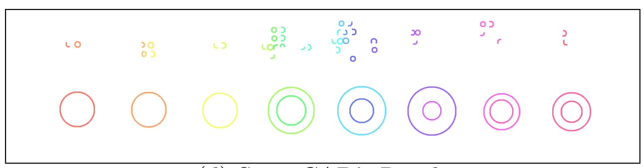


(a) Original Image

(b) Ground-Truth Image

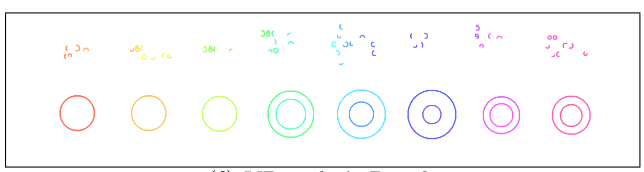


(c) Wenyin's Result



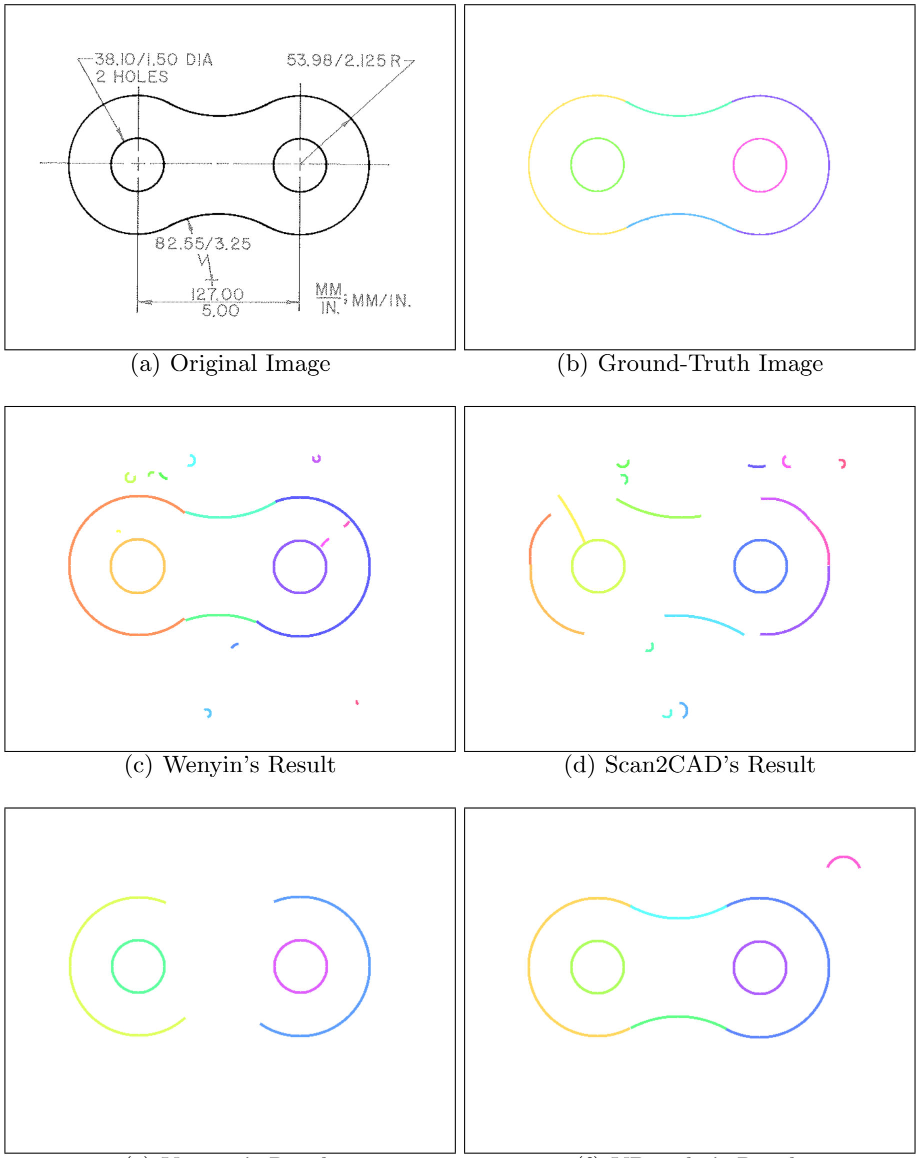
(d) Scan2CAD's Result

(e) Vectory's Result



(f) VPstudio's Result

Fig. 3. Results of the four participating methods on the second test image. All methods correctly found the circles, but also produced many false alarms originating from the text in the annotations, except the Vectory software which seems to have removed the text parts prior to arc segmentation.

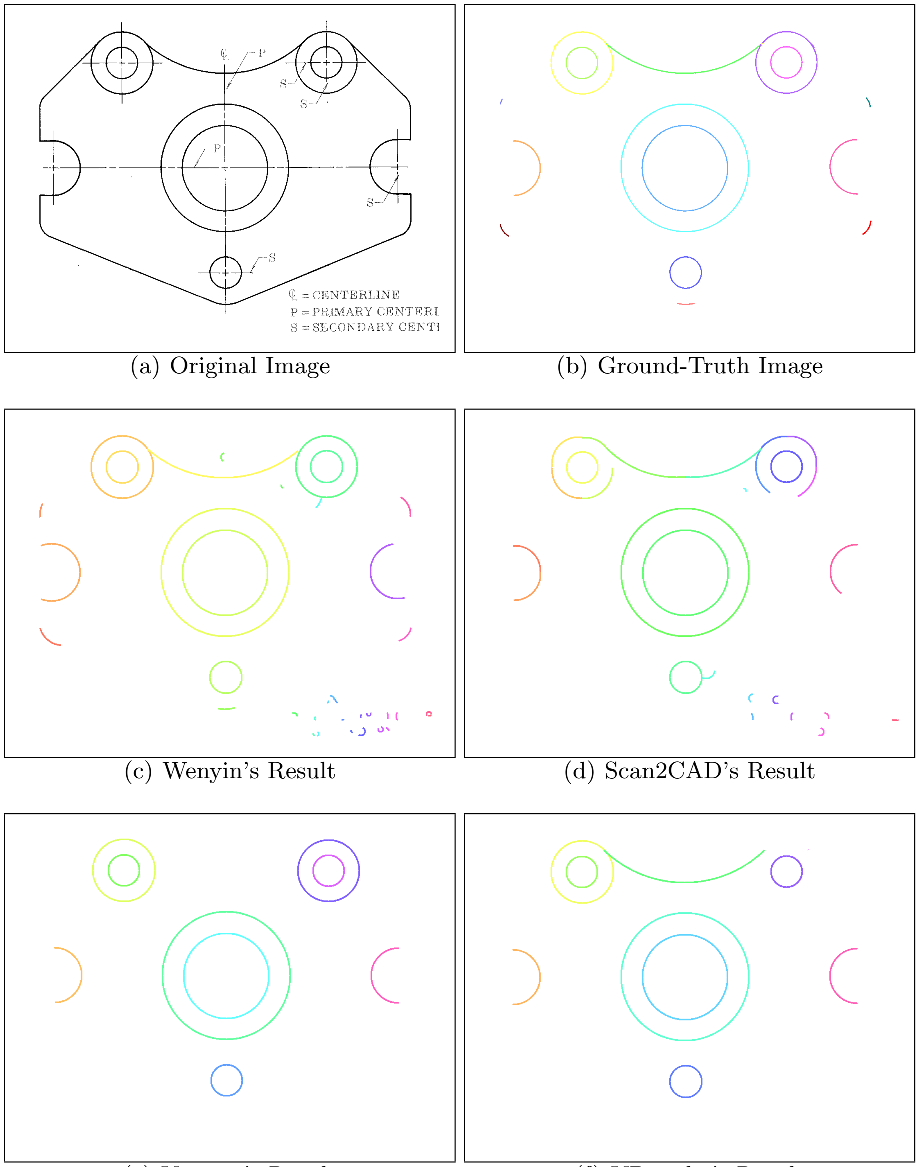


(a) Original Image (b) Ground-Truth Image

(c) Wenyin's Result (d) Scan2CAD's Result

(e) Vectory's Result (f) VPstudio's Result

Fig. 4. Results of the four participating methods on the third test image. VPstudio software had the best results in this case, since it was the only method that correctly segmented the two concave curves.

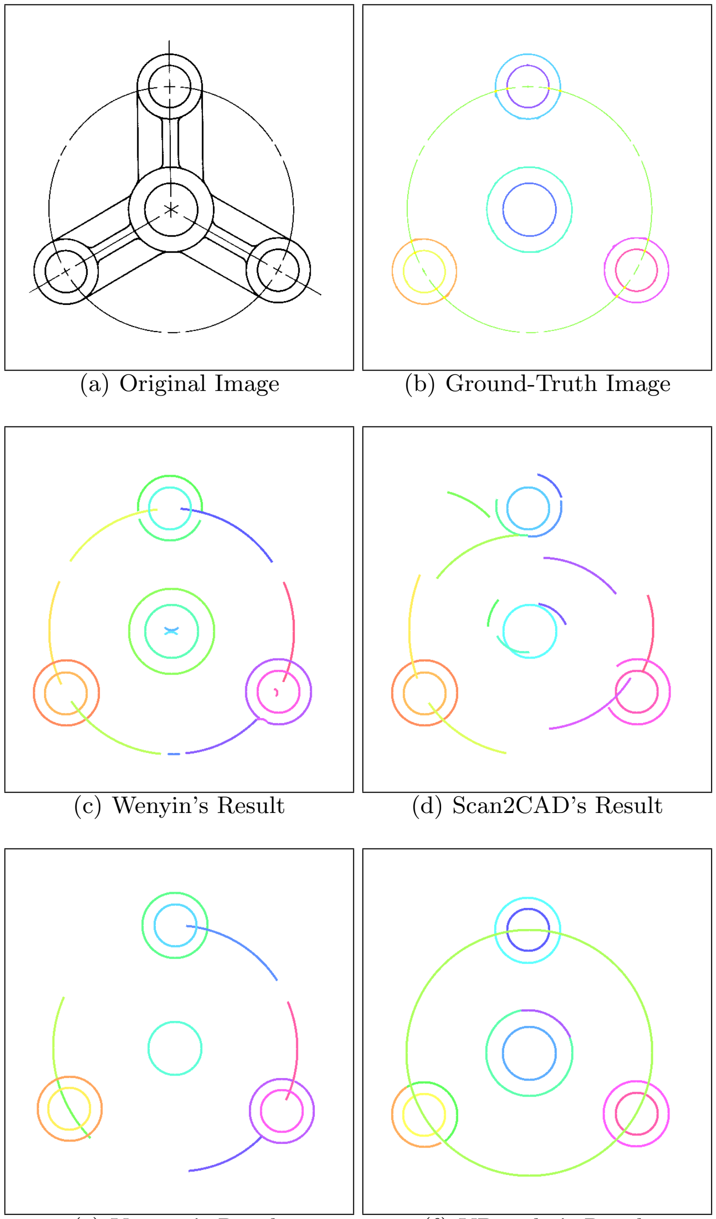


(a) Original Image (b) Ground-Truth Image

(c) Wenyin's Result (d) Scan2CAD's Result

(e) Vectory's Result (f) VPstudio's Result

Fig. 5. Results of the four participating methods on the fourth test image. Liu Wenyin's method had the best results for this image since it was the only method that correctly found the curved corners in the image.

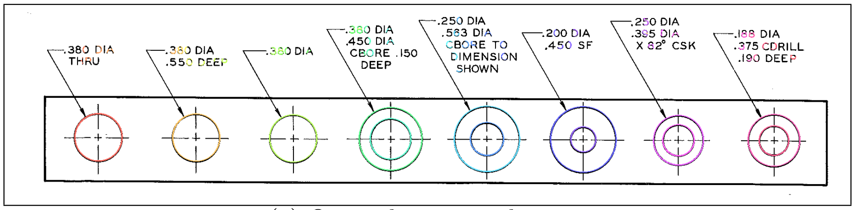


(a) Original Image (b) Ground-Truth Image

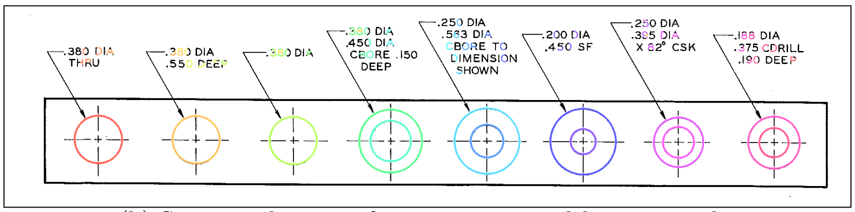
(c) Wenyin's Result (d) Scan2CAD's Result

(e) Vectory's Result (f) VPstudio's Result

Fig. 6. Results of the four participating methods on the fifth test image. Both VPstudio and Liu Wenyin's method had comparable results in this case that were better than those of Vectory and Scan2CAD.



(a) Original segmented image



(b) Segmented image after setting arc width to 10 pixels

Fig. 7. (a) The segmentation result of VPstudio on a test image. Despite the algorithm working very well in segmenting the arcs, the evaluation result reported that all 13 arcs were oversegmented in this image, since the results were supplied with a constant line width of one pixel. (b) The segmentation result after setting arc width to 10 pixels. The evaluation result for this image reported no segmentation errors.

Table 1 shows the vectorial score obtained by all participating methods on the test images. The results were obtained by using a relative threshold of 0.1 and an absolute threshold of 100 pixels. This implies that a segmentation error was considered significant only if the number of in-correctly segmented pixels was either larger than 10% of the pixels belonging to an arc or was larger than 100 pixels in total. The results show that all the algorithms over-segmented the arcs. This happened when all pixels belonging to an arc were not assigned to the arc by the algorithm. One major reason for this was that the VEC-files for the commercial systems were supplied with a constant line width of one pixel for all the arcs. An example showing the segmentation results of the VPstudio software on a test image is shown in Figure 7(a). Evaluation result for this image reported that all 13 arcs were over-segmented. To see the influence of this problem, we re-ran the evaluation using a constant line width of 10 pixels for all systems. Since we use only the foreground pixels while ignoring the background pixels, setting the arc width to 10 pixels has the effect of actually ignoring the arc width. The effect of setting the arc width of all arcs to 10 pixels for the example image of Figure 7(a) is shown in Figure 7(b). It can be seen that all the pixels belonging to an arc are now correctly assigned to that arc. Table 2 shows the evaluation results by ignoring the arc width. This table shows that most of the over-segmentations were due to the small line thickness supplied by the systems.

Table 1. Different types of errors made by each algorithm on the test images. The column labels are: total oversegmentations (T_o), total undersegmentations (T_u), oversegmented components (C_o), undersegmented components (C_u), missed components (C_m), false alarms (C_f)

Algorithm	T_o	T_u	C_o	C_u	C_m	C_f
Wenyin's method	21	8	13	6	1	93
Scan2CAD	72	9	48	7	9	64
Vectory	54	9	43	9	14	0
VPstudio	55	4	49	3	8	64

Table 2. Different types of errors made by each algorithm on the test images when arc width was set to a constant value of 10 pixels in the output of all algorithms.

Algorithm	T_o	T_u	C_o	C_u	C_m	C_f
Wenyin's method	17	9	9	6	1	94
Scan2CAD	36	10	20	6	9	66
Vectory	35	13	26	9	13	1
VPstudio	7	5	7	4	8	62

The results show that Wenyin's method and VPstudio software worked very well in segmenting arcs from the images and did uniformly better than the other two systems on most of the performance measures. The number of false alarms were high for these systems because they did not remove text parts in the images prior to arc recognition. From that aspect Vectory software performed the best by removing all textual components from the image, thereby resulting in no false alarms. Interestingly, for the test image shown in Figure 3, evaluation of the output of Vectory software reported all 13 arcs as over-segmented. A closer look revealed that the Vectory software also did a skew correction of the image, thereby slightly moving all circles from their original position. This resulted in all circles reported as over-segmented. Liu Wenyin's method had the least number of missed errors. Inspection of the results revealed that most of the missed error in commercial systems originated from ground-truth arcs consisting of round corners as in Figure 5.

6 Conclusion

This paper presented a summary of the GREC 2007 arc segmentation contest. We described the pixel-accurate color-based representation of arc segmentation that was used in the competition along with a vectorial score for measuring arc segmentation accuracy. The vectorial score enables us to evaluate different aspects of an arc segmentation algorithm. One research algorithm by Liu Wenyin and three commercial systems namely Scan2CAD, Vectory, and VPstudio were presented for participation in the contest. Results showed that Wenyin's method and VPstudio out-performed the other two systems, whereas the performance of Wenyin's method and that of VPstudio software was not significantly different from each other.

Acknowledgments

This work was partially funded by the BMBF (German Federal Ministry of Education and Research), project IPeT (01 IW D03).

References

1. Wenyin, L.: The third report of the arc segmentation contest. In: Proc. 6th IAPR Workshop on Graphics Recognition, Springer LNCS 3926, Barcelona, Spain (2005) 358–361
2. Wenyin, L.: Report of the arc segmentation contest. In: Proc. 5th IAPR Workshop on Graphics Recognition, Springer LNCS 3088, Barcelona, Spain (2003) 364–367
3. Wenyin, L., Zhai, J., Dori, D.: Extended summary of the arc segmentation contest. In: Proc. 4th IAPR Workshop on Graphics Recognition, Springer LNCS 2390. (2001) 343–349
4. Antonacopoulos, A., Gatos, B., Bridson, D.: ICDAR 2007 page segmentation competition. In: Proc. 9th Intl. Conf. on Document Analysis and Recognition, Curitiba, Brazil (2007) 1279–1283
5. Maergner, V., Abed, H.E.: ICDAR 2007 - arabic handwriting recognition competition. In: Proc. 9th Intl. Conf. on Document Analysis and Recognition, Curitiba, Brazil (2007) 1274–1278
6. Shafait, F., Breuel, T.M.: Document image dewarping contest. In: 2nd Int. Workshop on Camera-Based Document Analysis and Recognition, Curitiba, Brazil (2007) 181–188
7. Wenyin, L., Dori, D.: A protocol for performance evaluation of line detection algorithms. Machine Vision and Applications: Special Issue on Performance Characteristics of Vision Algorithms **9** (1997) 240–250
8. Shafait, F., Keysers, D., Breuel, T.M.: Pixel-accurate representation and evaluation of page segmentation in document images. In: 18th Int. Conf. on Pattern Recognition, Hong Kong, China (2006) 872–875
9. Wenyin, L., Dori, D.: Incremental arc segmentation algorithm and its evaluation. IEEE Trans. on Pattern Analysis and Machine Intelligence **20** (1998) 424–431