IJDAR (1998) 1: 37–42

International Journal on IJDAR
Document Analysis and Recognition

Document image similarity and equivalence detection

Jonathan J. Hull

Ricoh California Research Center, 2882 Sand Hill Road, Suite 115, Menlo Park, CA 94025, USA; e-mail: hull@crc.ricoh.com

Received July 13, 1997 / Revised November 29, 1997

Abstract. An algorithm is presented for determining the similarity and equivalence of document images. Features extracted from the CCITT fax-compressed representations of two images are compared to determine their visual similarity and whether they are equivalent (i.e., scanned from the same original). Pass codes in the compressed data are used as features. A fixed grid is imposed on the image and a feature vector is derived from the number of pass codes in each grid cell. The features vectors are compared to locate a group of documents that are visually similar to the input image. The equivalence of two documents is determined by applying the Hausdorff distance to the two-dimensional arrangement of pass codes in small patches of each image.

Key words: Document image similarity – Document image equivalence – Similarity detection algorithm – Equivalance detection algorithm

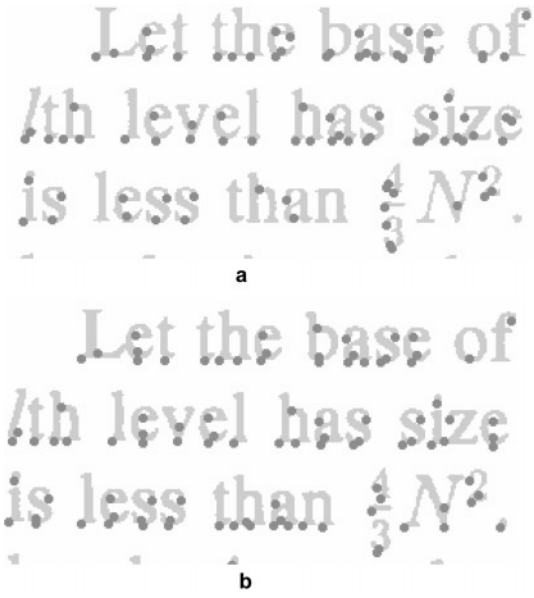

Fig. 1a,b. Pass codes in a CCITT group 4 fax coded image. Scanned original **a** and scanned photocopy of the same document **b**

1 Introduction

A document image matching algorithm is given a page image and it determines whether a visually similar page is contained in an image database. Such an algorithm is useful for an automatic filing application in which scanned document images are stored in directories that contain visually similar documents. For example, business letters might be stored separately from scientific papers. Another application determines whether a specific document image is "equivalent" to another image in a large database. Two document images are said to be equivalent if they were scanned from substantially the same original or a photocopy of it. This process has also been called *duplicate detection* [1].

Several methods have been used for detecting equivalent document images. These include matching features extracted from text such as the sequence of word lengths [2] or character shape codes [1]. Both of these approaches calculate features from words of text in a document image. This paper proposes an alternative approach in which the features are extracted from text-like areas in a compressed image format. An explicit segmentation into words is not required.

This paper proposes a combination of feature extraction from CCITT group 4 fax-compressed images with a matching algorithm based on the Hausdorff distance. The feature extraction takes advantage of information represented in the compressed format. The group 4 standard is a two- dimensional coding scheme in which runs of black (or white) pixels are coded with respect to runs of the same color on an adjacent row. A format known as a "pass code" is used for runs that have no corresponding adjacent run. This attaches pass codes to the bottoms of a large percentage of connected components. Pass codes were previously used to determine the skew of a document image [7].

The rest of this paper contains a short explanation of the CCITT format and the algorithm for document

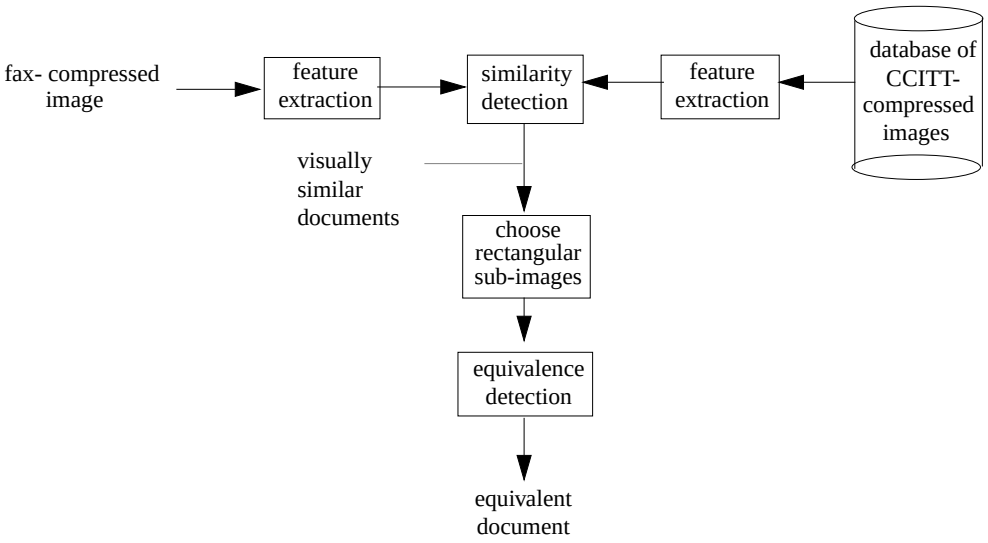

Fig. 2. Document image similarity and equivalence detection algorithm

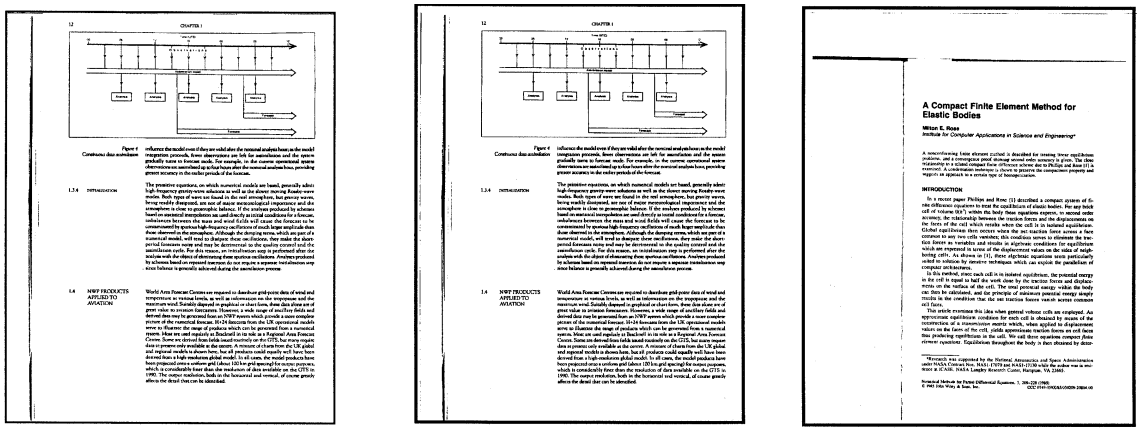

Fig. 3. Example of input image and feature vector used for similarity detection **a**. Images and feature vectors **b** and **c** that minimized Euclidean distance to **a**

matching. An experimental investigation is reported that demonstrates the effectiveness of the similarity and equivalence detection steps.

2 CCITT fax encoding

The CCITT group 3 fax encoding of a binary image uses a combination of a one-dimensional coding scheme applied to every kth line (typically, k equals 4) with a two-dimensional coding method applied to the other lines. The 2D technique codes black and white runs in a given row with respect to corresponding runs in the previous row. In the group 4 standard, all lines are coded two-dimensionally. The first line in an image is coded with respect to an all-white reference line.

The 2D coding algorithm uses three coding modes: horizontal, vertical, and pass. A pass code is used when a run on one line has no corresponding run in an adjacent line. This indicates the termination of white or black components. For example, many characters have white pass codes attached to holes and black pass codes attached to the bottoms of strokes. Approximately 80–90% of the components have an attached pass code. This characteristic is illustrated in the portion of the document image shown in Fig. 1. The pass codes (i.e., the $x - y$ location of each pass-coded run) extracted from a scanned original are shown in Fig. 1a and the pass codes extracted from a scanned photocopy of the same document are shown in Fig. 1b.

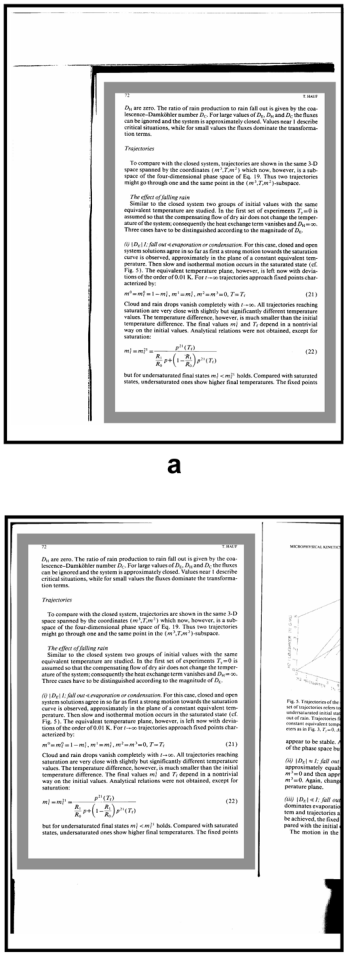

Fig. 4. Example of data areas on E008 **a** and S008 **b**

3 Proposed algorithm

The proposed algorithm for document image matching is shown in Fig. 2. First, the $x-y$ locations of pass codes are extracted from an input fax- encoded image. Then the number of pass codes in each cell of a fixed grid are determined. A group of visually similar documents is located by calculating the Euclidean distance between these feature vectors. A fixed number of images with the smallest distances provide a group of documents that are likely to contain an equivalent image, if one exists in the database.

Equivalent documents are detected by locating a patch (e.g., the upper left 1-inch by 1-inch region of the first paragraph) in each image that could be the same in both documents. The $x-y$ locations of the pass-coded runs in each patch are compared using the Hausdorff distance. Two patches with a Hausdorff distance below a threshold are assumed to have been derived from different scanned versions the same document page. The objective is to locate different versions of the same document page, if it exists in the database. False positives may occur since only a small portion of two page images are actually compared. This technique has been used successfully without a similarity detection algorithm [3].

A modified Hausdorff distance has been proposed for locating models of objects in two-dimensional scenes [4]. This modified distance measures the percentage p_m of model points that are within d_m pixels of some image point for some given translation of the model. It also measures the percentage p_i of image points that are within d_i pixels of some model point, for the same translation. An efficient implementation of the modified Hausdorff distance has also been developed [6].

Figure 1 helps explain why the Hausdorff distance applied to pass code locations is effective in locating equivalent document images. Even though a number of pass codes occur in one image and not the other, if they do occur in both images they are very likely to be close to one another, for some $x-y$ translation that registers the documents. This is because of the natural constraint provided by the content of the text passage (i.e., the sequence of characters it contains), as well as the font, point size, pitch, and spacing in a typeset passage.

4 Experimental results

A series of experiments were performed that investigated the performance of the similarity and equivalence detection steps. The 979 images on the first University of Washington CD-ROM [5] were used to test the algorithm. Each image on the CD-ROM is labeled with a four character identifier, e.g., A001, A002, and so on.

There are 146 pairs of images (i.e., 292 individual images) on the CD-ROM that were scanned from different generation photocopies of the same document. Of these, 125 pairs contain one document labeled with an "E" and another labeled with an "S." For example, E001 was scanned from a first generation photocopy and S001 was scanned from the original version (i.e., zero generation photocopy) of the same original document. All the 125 images labeled with an initial "E" were scanned from a first generation photocopy. Of the 125 images labeled with an initial S, 10 were scanned from a first generation photocopy and 115 were scanned from the original document.

The other 21 pairs contain one document labeled with an "IG" and another labeled with an "I." The "IG" images were scanned from first generation photocopies and the "I" images were scanned from second generation photocopies.

4.1 Similarity detection

The objective of testing the similarity detection algorithm was to determine whether it finds the corresponding member of a pair among its ten best choices when it is given a document labeled E, S, IG, or I00 as input. Also, the vector size needed to obtain the best performance was investigated. Figure 3 shows an example of a 4-element feature vector extracted from a 2×2 grid imposed on a document image. The two images in the database with feature vectors that are closest (i.e., minimize the Euclidean distance) to the given document are

Table 1. Performance of similarity detection in finding duplicate documents

Pre-process	Vect. size	Choice										Cum. %
		1	2	3	4	5	6	7	8	9	10	
None	2×2	51	12	10	7	3	3	3	1	5	2	33%
	3×3	71	14	9	6	2	4	4	2	2	1	39%
	4×4	92	3	2	6	1	4	3	4	3	3	41%
	5×5	99	6	9	4	2	2	6	3	1	1	46%
Page boxes	2×2	149	25	12	5	4	6	3	2	3	6	74%
	3×3	205	12	7	3	3	2	3	0	1	3	82%
	4×4	220	9	6	2	0	2	2	2	2	1	84%
	5×5	228	10	3	4	3	2	2	0	0	0	86%
Comp. sizes	2×2	215	24	12	7	4	2	6	1	3	1	94%
	3×3	256	13	5	3	2	0	1	2	0	1	97%
	4×4	267	7	4	1	4	0	0	2	0	0	98%
	5×5	271	6	4	1	1	0	1	1	0	0	98%

also shown. In this case the similarity detection procedure located the pair of matching documents (E03J, S03J).

The performance of the similarity detection technique was calculated using grid sizes 2×2, 3×3, 4×4, and 5×5. This provided feature vectors that have 4, 9, 16, and 25 integers, respectively. The feature vector from each of the 979 images on the CD-ROM was compared to the feature vectors for all the other 978 images to obtain the results summarized in Table 1. A "correct" answer is returned if, when given a document for which a duplicate exists in the database, it finds that duplicate among the N images with the minimum Euclidean distance. The absolute number of correct choices in each of the first $N = 10$ positions is shown as well as the overall percent correct in the top 10.

Three methods of preprocessing were used. In the simplest case (i.e., no preprocessing), the $x - y$ coordinates of all the pass-coded runs in every document were used. With a 5×5 grid, this successfully located 46% of the members of all duplicate pairs of documents in the top ten choices. The second type of preprocessing admitted only pass codes that were contained within the data area of each page. This eliminates extraneous pass codes from adjacent pages and from border regions that were scanned inadvertently. The data area was determined by taking the union of the bounding boxes for the zones given in the truth files for the database. Figure 4 shows an example of the data areas on two pages extracted with this procedure. This preprocessing could be useful in a controlled scanning situation, e.g., when a document feeder is used, in which extraneous data such as that shown in Fig. 4 could be eliminated. The results show that this preprocessing significantly improved performance: 86% of the duplicate images were located in the ten best choices when a 5×5 grid was used.

The third type of preprocessing used the x and y values of the lowest point on selected connected components as features. Only connected components that were approximately the size of characters were used (i.e., $5 \leftrightarrow 70$ pixels wide and $10 \leftrightarrow 70$ pixels high). Other researchers have shown how this information could be extracted from images compressed in a similar format [8]. In the experiments reported here, the connected component information was calculated from the uncompressed image. The results show that this preprocessing was successful in finding the correct document among the first ten choices in up to 98% of all cases. This varied between 94% and 98% for the different vector lengths. Of the 7 images that could not be matched when a 5×5 grid was used, 4 of them (E02H, E02I, S02H, and S02I) were caused by documents that were scanned at different magnifications. These errors are understandable since this technique was not designed to handle scale changes. Two of the errors (IG0G and I00G) apparently occurred because one photocopy was made with the cover closed and the other without the cover closed.

4.2 Equivalence detection

Another set of experiments explored the ability of the proposed algorithm to locate images scanned from different photocopies of the same document in a large database. A one-inch square patch was extracted from the upper left corner of the first body text zone of 800 document images on the CD-ROM. The patch size (one inch square) was chosen so that it covered at least two (preferably three) lines of text. This takes into account the two-dimensional layout of a number of characters. Figure 5 shows an example of the pass codes extracted from such a patch. The upper-left corners were located from the truth files. The 800 images were those that contained at least one zone classified as "text-body", i.e., were at least one inch by one inch. The other 179 images either contained no body text or the largest body text zone was smaller than one inch square.

This was designed to be a stress test for the equivalence testing algorithm. That is, the previous results showed that the equivalence detection algorithm may only have to be executed on as few as 10 image hypotheses chosen from an initial set of 979 documents. Successful equivalence detection on 800 images implies that the combined approach of similarity detection followed by equivalence detection could also work well on a larger initial set of documents.

The 800 images contain 266 (i.e., 133 pairs) that were scanned from different photocopies of the same original document. This includes 114 images labeled E, 114 im-

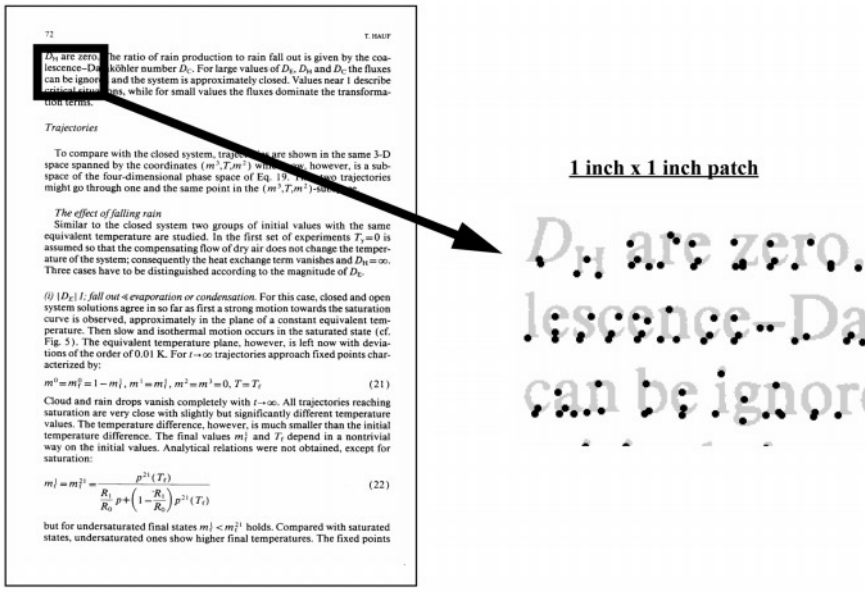

Fig. 5. Pass code locations extracted from a document image

Table 2. Equivalence detection performance

Test condition	N	% corr.	% error (false positives)	% missed
Original: original	800	100%	0%	0%
Duplicate pairs	266	95%	0%	5% (12 images)

ages labeled S, 19 images labeled I, and 19 images labeled IG. For any image, there is at most one other equivalent version in the database.

The 2D arrangement of pass codes in each of the 800 test patches was matched to the 2D arrangement of pass codes in all the 800 patches in the database. The parameters used were $p_m = 70$, $d_m = 4.0$, $p_i = 50$, and $d_i = 4.0$.

The results of this experiment are shown in Table 2. This shows that every patch was uniquely matched to itself (100% correct, 0% error). Also, 95% of the patches extracted from duplicate images in the database were also correctly located. That is, there was a 5% miss rate. This indicates that it is possible to derive a unique identifier for a complete page of English text from a relatively small amount of image data (one square inch) extracted from it. Also, this representation is tolerant to the noise that typically occurs when a document is photocopied.

The run time of the matching algorithm was measured to be about 10 minutes for each patch on the given database of 800 documents. This was calculated on a 70 MHz Sun Sparcstation 20. Since the time requirement is roughly linear in the number of images, approximately 7.5 s would be needed to apply the equivalence detection routine to 10 documents output by the similarity detection step.

An analysis of the errors showed that four of them were caused by nonlinear distortions such as those shown in Fig. 6 which can occur when copying pages near the binding of a book. Four of the errors were caused by scale differences between the original and the photocopy (see Fig. 7 for an example). The other four errors were caused by several reasons that might be corrected by further adjustment of the parameters.

5 Conclusions and future directions

A hierarchical algorithm for document image matching was proposed that used pass codes in fax-compressed document images as features. A group of similar images is first determined with a feature vector that counts the number of pass codes in each cell of a fixed grid. Equivalent images are then located by applying a modified Hausdorff distance to the documents returned by the first step.

Areas for improving the similarity detection algorithm include incorporation of structural information about the segmentation of a document into such zones as text, graphics, photographs, etc. Clearly, the overall accuracy of the similarity detection method depends on the heterogeneity of document formats in the database. Performance can be expected to degrade as the number of documents with the same format (e.g., two column text of the same line width) increases. An obvious area for improvement is automatic thresholding of the number of images returned by similarity detection.

The experiments with the equivalence detection method assumed corresponding image patches could be located in two equivalent images. An automatic technique for calculating this from compressed data should be developed. Such a method could also extract and compare multiple patches from two images. This would increase the accuracy of the present method and reduce the probability of false positives.

Future work on this approach should include application to larger databases of document images with distinctly separate training and testing sets. The effect on performance of different sizes for the grid used in similarity detection and the patch used in equivalence detection should also be investigated.

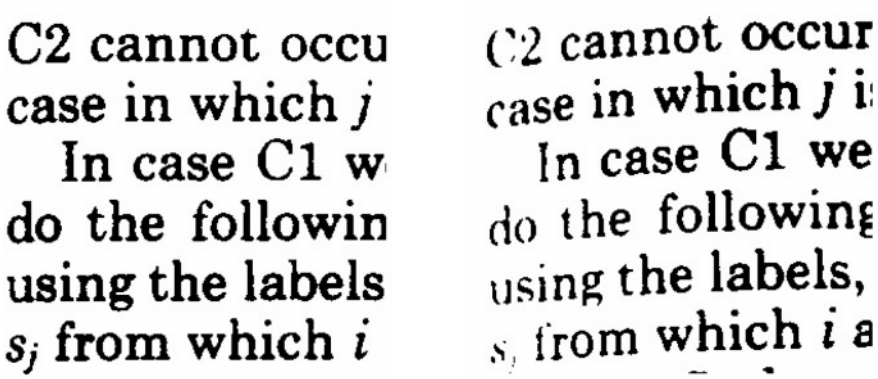

Fig. 6. Error caused by nonlinear distortion

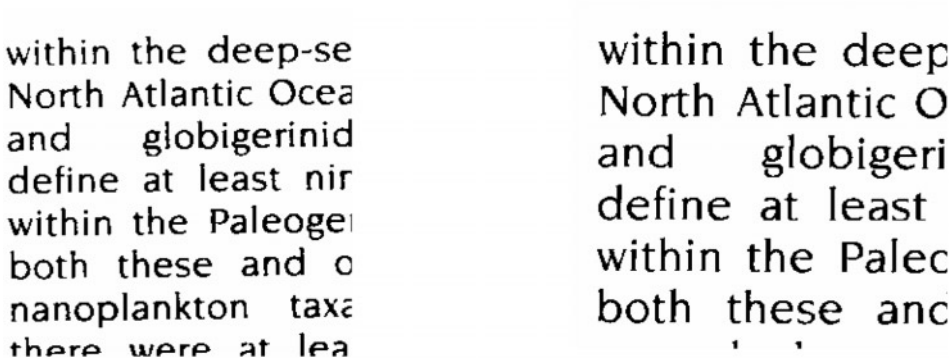

Fig. 7. Error caused by scale difference between original and photocopy

References

1. D. Doermann, H. Li, O. Kia: The detection of duplicates in document image databases. Fourth International Conference on Document Analysis and Recognition, Ulm, Germany, August 18–20, 1997, pp 314–318
2. J.J. Hull: Document image matching and retrieval with multiple distortion-invariant descriptors. In: A. Lawrence Spitz, A. Dengel (eds) Document Analysis Systems. Singapore, World Scientific, 1995, pp 379–396
3. J.J. Hull: Document matching on CCITT group 4 compressed images. SPIE Conference on Document Recognition IV, San Jose, CA, February 12–13, 1997, pp 82–87
4. D. Huttenlocher, G. Klanderman, W. Rucklidge: Comparing images using the Hausdorff distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(9), 1993, pp 850–863
5. I.T. Phillips, S. Chen, R.M. Haralick: CD-ROM document database standard. Proceedings of the Second International Conference on Document Analysis and Recognition, Tsukuba Science City, Japan, October 20–22, 1993, pp 478–483
6. W. Rucklidge: Efficient computation of the minimum Hausdorff distance for visual recognition. TR94-1454, Cornell University, Department of Computer Science, September 8, 1994
7. A.L. Spitz: Skew determination in CCITT group 4 compressed document images. Proceedings of the Symposium on Document Analysis and Information Retrieval, Las Vegas, March 16–18, 1992, pp 11–25
8. H. Takeda: Image Component Labeling from MR Codes. Transactions of the Institute of Electronics, Information and Communication Engineers, vol. J76-D-II (11), 1993, pp 2341–2348 (in Japanese)

Jonathan J. Hull received the B.A. degree in computer science and statistics and the M.S. and Ph.D. degrees in computer science from the State University of New York at Buffalo (SUNYAB) in 1980, 1983 and 1987, respectively.

He currently heads the Document Analysis Group at the Ricoh California Research Center in Menlo Park, California. From 1987 to 1994 he was a member of the research faculty of the Department of Computer Science at SUNYAB and was also the Associate Director of the Center of Excellence for Document Analysis and Recognition (CEDAR). His research interests include document analysis, computer vision, pattern recognition, and information retrieval.

Dr. Hull is a member of the ACM, the IEEE Computer Society, and is an Associate Editor of Pattern Recognition and the IEEE Transactions on Pattern Analysis and Machine Intelligence. He served as co-chair of the Second IAPR Symposium on Document Analysis Systems that was held in Malvern, Pennsylvania in October, 1996. He will co-chair the Program Committee of the Fifth International Conference on Document Analysis and Recognition that will be held in Bangalore, India in September, 1999.