Applied Intelligence 18, 257–270, 2003
© 2003 Kluwer Academic Publishers. Manufactured in The Netherlands.

Text Retrieval from Document Images Based on Word Shape Analysis

CHEW LIM TAN, WEIHUA HUANG, SAM YUAN SUNG, ZHAOHUI YU AND YI XU
School of Computing, National University of Singapore, 3 Science Drive 2, Singapore 117543

Abstract. In this paper, we propose a method of text retrieval from document images using a similarity measure based on word shape analysis. We directly extract image features instead of using optical character recognition. Document images are segmented into word units and then features called vertical bar patterns are extracted from these word units through local extrema points detection. All vertical bar patterns are used to build document vectors. Lastly, we obtain the pair-wise similarity of document images by means of the scalar product of the document vectors. Four corpora of news articles were used to test the validity of our method. During the test, the similarity of document images using this method was compared with the result of ASCII version of those documents based on the N-gram algorithm for text documents.

Keywords: document image analysis, text retrieval, similarity measure, document vector

1. Introduction

The Digital Library of the National University of Singapore archives various historical documents for public access and viewing on the web. The archived documents include century old newspapers and past student theses. To facilitate access to these documents, a text retrieval method based on text similarity is explored.

There are many ways to measure text similarity of documents. One way is to analyze the similarity of the documents' contents based on semantics but this needs a large amount of processing time. Another way is to use a statistical method without the need to understand the meaning of documents. A common statistical approach is the construction of document vectors based on the frequencies of words or N-gram character sequences. This method is easy to implement without too much processing time. Many researchers used this approach to classify document texts.

In applying text similarity measure on document images in our present application, it is necessary to use optical character recognition (OCR) to convert the images to textual information. However, OCR systems are not perfect especially for poor image quality, requiring expensive manual correction. In fact, an original plan of the library to OCR the entire archive set has proven to be overwhelmingly costly. An alternative approach to gauge text similarity from the document images directly by means of word shape analysis is thus proposed. The word shape analysis selects an image feature called the vertical bar pattern in order to construct document vectors.

The remainder of this paper is organized as follows. Section 2 surveys related works in text retrieval of electronic texts as well as document images. Section 3 describes the feature extraction process to extract vertical bar patterns and forming feature vectors from the document images. Section 4 presents the methods for measuring similarities between documents. Section 5 discusses experimental results that confirm the validity of the proposed model. Finally, conclusions and future work are given in Section 6.

2. Related Works

Over the past few decades, methods of categorisation and retrieval of machine-readable texts [1–3] have been proposed. They have relied on self-evident utility of words, sentences, and paragraphs for sorting, categorising, and retrieving texts. Furthermore, various means of suppressing uninformative words, removing prefixes,

suffixes, and endings, interpreting inflected forms, etc. have been developed. Depending on the application, these methods share a number of potential drawbacks: they require a linguist or a polyglot for initial set-up and subsequent tuning, they are vulnerable to variant spellings, misspellings, and random character errors, and they tend to be both language-specific and domain-specific.

The purely statistical characterisation of text in terms of word frequency or N-grams (sequences of N consecutive characters) [4] has been applied to text analysis and document processing, including spelling and error correction [5–11], text compression [12], language identification [13, 14], and text search and retrieval [15, 16]. Basing on this statistical characterisation, M. Damashek [17] has proposed a simple but novel vector-space technique that makes sorting, clustering and retrieval feasible in a large multilingual collection of documents.

Damashek's method does not rely on words to achieve its goal, and no prior information about the document content or language is required. It only collects the frequency of each N-gram to build a vector for each document and the processes of sorting, clustering and retrieval can be implemented by measuring the similarity of the document vectors. It is language-independent. A little random error only influences a small quantity of N-grams and will not change the total result. This method thus provides a high degree of robustness.

Text in document images is a more complicated matter for text retrieval. One common method is to convert it to machine readable text using optical character recognition (OCR) first and then use the usual text retrieval techniques. However, character recognition systems are not perfect and require human correction. Some works, however, are done to bypass the human correction to do retrieval of OCR degraded text [18] and simulated OCR output [19] with reasonable results. Furthermore, OCR requires a significant amount of processing time and is language-dependent. A typical system can only recognize one or several languages. We need to know the specific language in the document beforehand.

Instead of relying on OCR, another approach is to retrieve information based on the image content directly. This does not require language identification. Recently, several researchers have made such an attempt in a number of applications. For example, F.R. Chen and D.S. Bloomberg [20, 21] have described a method for automatically selecting sentences for creating a summary from a document image without recognition of the characters in each word. They build word equivalence classes by using a rank blur hit-miss transform to compare word images and use a statistical classifier to determine the likelihood of each sentence being a summary sentence. Hull and Cullen [22] have proposed a method to detect equivalent document images by matching the pass codes of document. They create a feature vector that counts the numbers of pass codes in each cell of a fixed grid in the image and equivalent images are located by applying the Hausdorff distance to the feature vectors.

Other researchers have also proposed methods to retrieve text directly from non-English document images. For instance, Y. He et al. [23] have proposed an index and retrieval method for Chinese document images based on a stroke density code. Language classification of multilingual documents is another field having been researched. A.L. Spitz et al. [24, 25], C.Y. Suen et al. [26] and C.L. Tan et al. [27] have developed systems to identify Latin-based languages, Han-based languages and other languages using the character shape coding [28].

A local extrema point detection method was initially suggested for handwriting characters recognition [29]. This feature can effectively reflect the property of a word, and it is relatively invariant to touching and broken characters. Extending the original idea, we apply the local extrema points detection and form the so-called vertical bar patterns from these points. Experimental results show that this method is also relatively invariant to changing of fonts and styles, and to the degradation of document qualities.

3. Feature Extraction

Figure 1 outlines the steps in gauging the similarity of document images based on content. In the preprocessing stage, document images are segmented into word units and filters are applied to remove punctuation marks and small noise. In the feature extraction stage, local maximum points and local minimum points within these word units are identified. After these local maximum and minimum points are paired up, a list of vertical bars is obtained. After classification of these vertical bars (to be explained in Section 3.3), each word unit is converted into a list of symbols consisting "d", "m" and "q", or what we call the vertical bar pattern. A feature vector can be obtained from each

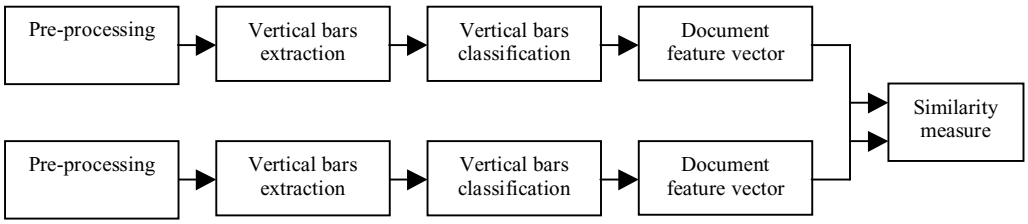


Figure 1. Gauging the similarity of document images based on content.

document image. Feature vectors are used to calculate the similarity of document images by calculating their scalar products.

3.1. *Pre-processing*

Document images can be segmented into a number of word units through layout analysis. A word unit contains connected components. These connected components are either character objects, noise or punctuation marks. A character object may be a single character, or part of a broken character, or characters touching together. Types of character objects are not important, since we will do the feature extraction at the word level. But noise and punctuation marks are not considered as parts of the content of the document and thus need to be removed.

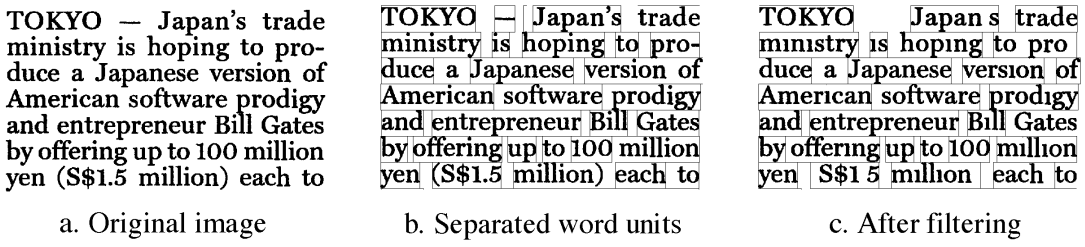
In general, the height of a punctuation mark is less than that of a character. Furthermore, dots and straight lines often have higher pixel densities than character objects. So, when connected components have been retrieved, we can apply filters to remove objects with smaller width or height or higher pixel density than predetermined thresholds. Most of the punctuation marks and noise will be removed in this manner. For removing brackets, we notice that brackets have an outstanding height in a text-line, and thus will be filtered by a predetermined threshold. Note that in some of the Latin languages such as French and German, there are accents and special characters. These accents and special characters appear as small connected components in document images. They are thus also removed by the filters. The removal of accents should not seriously affect the similarity measure just as the absence of accents in some printed documents due to inadequate printing capabilities does not reduce human readability of the documents. This however can be confirmed in a future study.

Figure 2 shows the result of document image after pre-processing. Item (a) is the original document image. Item (b) outlines the segmented word units and item (c) shows the word units after removal of punctuations and noises. Notice that the dots belonging to characters "i" and "j" are also removed. However this is not a problem, since the vertical bar patterns for these characters depend on their major part, and the classification can still be done properly.

3.2. *Vertical Bars Extraction*

The formal definition of local maximum and local minimum in mathematics is given below, and its graphical representation is given in the Fig. 3.

Definition 1 (*The first derivative tests*). Given an arbitrary curve f(x), and two open intervals on the curve (a, c) and (c, b):



a. Original image b. Separated word units c. After filtering

Figure 2. Extraction of word units.

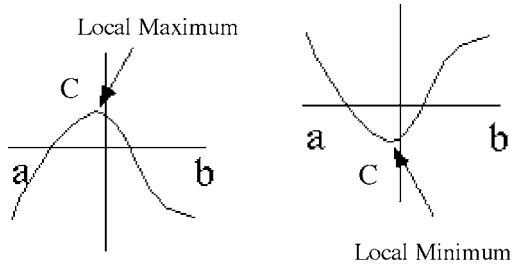


Figure 3. Graphical representation for definition of local extrema.

If $f'(x) < 0$ on (a, c), and $f'(x) > 0$ on (c, b) then f has a *local minimum* at $x = c$.
If $f'(x) > 0$ on (a, c), and $f'(x) < 0$ on (c, b) then f has a *local maximum* at $x = c$.

Each alphabetic character has a number of local maxima as well as a number of local minima. If we pair up these local maxima and minima, each pair forms a vertical bar whose top is a local maximum and bottom is a local minimum. Since a word is made up of characters, it can then be represented by a list of vertical bars contributed by its characters.

As vertical scan lines traverse across the word from left to right, top-most and bottom-most black pixels are recorded as the maximum and minimum points for visited lines. An algorithm is designed to detect local extrema points by keep tracking of the increasing and decreasing trends of the maximum and minimum points between neighbouring lines. Some glitches may exist along a horizontal image edge, they are detected and ignored through simple edge smoothing to ensure the validity of the local extrema points detected. Figure 4 illustrates the cases where edge smoothing is needed.

We are concentrating on the effective vertical bars only, and so short vertical bars are treated as noise. A length filter is applied to remove these noise bars. The threshold value for the length filter is calculated by the product of the average bar length and a predetermined percentage. Figure 5(a) illustrates the local extrema points detected for sample word "huge". Note that the number of local maxima points and the number of local minima points are not equal, thus some of the local extrema points are shared among vertical bars within the same character object. Figure 5(b) shows the list of vertical bars extracted by pairing up local maxima points and local minima points.

3.3. *Vertical Bars Classification*

To make the vertical bars comparable between documents regardless of their horizontal position, classification need to be done according to the vertical

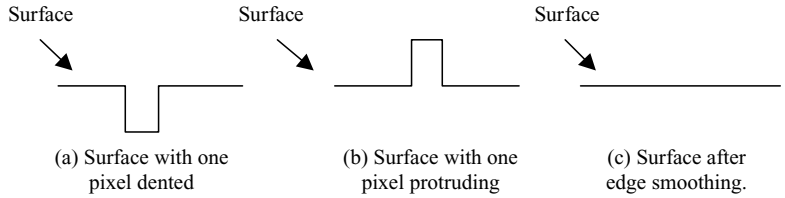


Figure 4. Word edge smoothing. (a) Missing pixel, (b) extra pixel, (c) edge after smoothing.

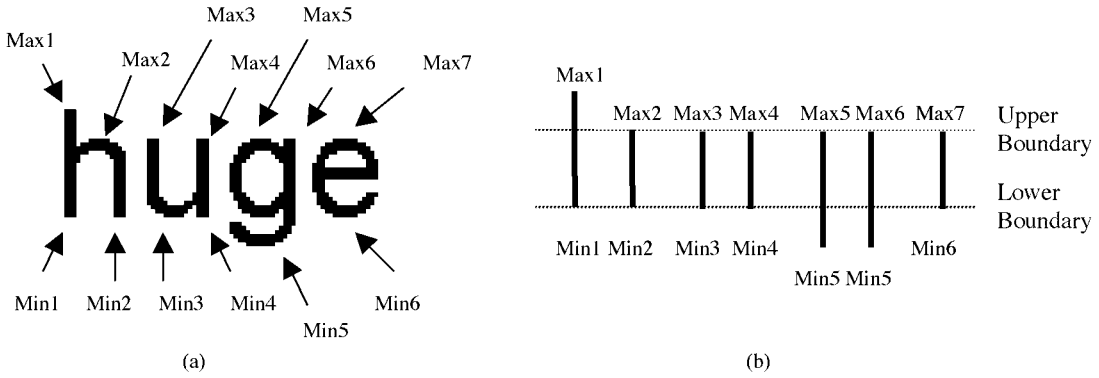


Figure 5. Vertical bar detection. (a) Finding local extrema points, (b) vertical bars extracted with boundary lines indicated.

bars' length and vertical position. By locating the upper boundary and lower boundary lines, we can divide the whole word into three vertical zones. Then vertical bars that protrude into the upper zone are in class "d", vertical bars that protrude into the lower zone are in class "q", and vertical bars that are only inside the middle zone are in class "m". There is a special case where the vertical bar pattern for character "f" protrudes into both upper zone and lower zone for some of the font types such as Lucida Sans, while for most of the font types the vertical bar pattern for character "f" only protrudes into the upper zone. To remove ambiguity we put the vertical bar pattern for "f" into class "d", and thus there is no overlapping between class "d" and class "q" vertical bars. Figure 5(b) indicates the two boundary lines detected. In this way, a word can be represented as a list of symbols consisting of "d", "m" and "q", where each symbol indicates a vertical bar from the corresponding class. For example the word "huge" is converted into vertical bar pattern "dmmmqqm".

3.4. Document Feature Vector

A hash table is created to keep track of the frequencies of all the vertical bar patterns being studied. Each hash table can be treated as a vector, so called the document vector. Each entry in the vector records a specific vertical bar pattern and its number of occurrences. Every time a vertical bar pattern is picked, the number of occurrences of the entry corresponding to the vertical bar pattern is increased by one.

The occurrence frequency of each vertical bar pattern is normalised by dividing it by the total number of occurrences of all extracted vertical bar pattern. This means that the absolute number of occurrences will be replaced with the relative frequencies of corresponding vertical bar patterns. The reason for doing this is that similar texts of different lengths after this normalisation will have similar document vectors. The feature vector for document m can be modelled as

$$X_m = x_{m1} x_{m2} \cdots x_{mN}$$

where N is the total number of distinct vertical bar patterns in the document m. Thus X_m is an N dimensional vector, and each element in the vector is defined as

$$x_{mi} = \frac{f(p_{mi})}{\sum_{j=1}^{N} f(p_{mj})}$$

where f is a function that returns the absolute frequency of a vertical pattern p_{mi} by looking up the hash table, and p_{mi} is the ith entry in the hash table created for the document m.

4. Similarity Measure

Document vectors for similar documents generally point in the same direction. The similarity score between two document vectors is defined as their scalar product divided by their lengths. A scalar product is calculated through summing up the products of the corresponding elements. This is equivalent to the cosine of the angle between two document vectors seen from the origin. So, the similarity between document images m and n will be

$$Similarity\,(X_m, X_n) = \frac{\sum_{j=1}^{J} x_{mj} x_{nJ}}{\sqrt{\sum_{j=1}^{J} x_{mj}^2 \sum_{j=1}^{J} x_{nj}^2}}$$

where X_m and X_n are the document vectors of documents m and n respectively, as defined in Section 3.4, J is the dimension of document vector, and $X_i = x_{i1} x_{i2} \cdots x_{iJ}$.

For each vertical bar pattern in the feature vector of the query document image, we look for corresponding entry in the feature vector of the database document image. If the pattern is found, then their product is obtained by multiplying their normalized frequencies, otherwise their product is simply zero since one of the feature vectors does not contain such a pattern. For all document images, the vertical bar patterns in the feature vectors are always in the form of strings containing "d", "m" and "q". Comparisons between these strings are straightforward to ensure the computational efficiency of the proposed method.

5. Experimental Results

Experiments were carried out to test the effectiveness of our image-based similarity measure in comparison with the traditional text-based similarity measure using the N-gram algorithm. If the results obtained by the image-based method are equally good or at least not much worse than the results obtained by the text-based method, it means we can effectively retrieve document images without resorting to full OCR, to avoid costly computations. To make the process simple, some preprocessing is done by de-skewing the images [30]. In the case where there are headlines and pictures or photographs, they are removed from the images. Four different corpora of document images were used in

the following tests to examine the ability of the proposed method to handle scanned document images in different qualities and different fonts etc. To create the ASCII versions of these documents as a means of benchmarking, an OCR system was used to extract the text from the images. The extracted texts were corrected by hand for any error from the OCR.

Corpus One (K01–K26) is made up of articles that were extracted from the Internet and were already electronically available. These ASCII text documents are labeled as O01–O26. The news articles were converted to images using the text input of Adobe Photoshop in 10-point Times New Roman font. These articles address four different topics, respectively. K01–K12 talk about economic crisis in Brazil, K13–K17 refer to personal computers, K18–K21 tell of scholarships and K22–K26 describe the news of a nuclear spy in the US. For each topic, we picked the first one of each group as the query article and thus K01, K13, K18 and K22 were selected. Similarity measures of all the articles in this corpus with the respective four query articles were made using the image-based and text-based methods. The results are summarized in Table 1 and Fig. 6.

From the testing with the first Corpus, it can be seen that a threshold may be set to decide whether a text is similar to a query article. The threshold lies somewhere in the region of 0.1 to 0.2. To further evaluate the performance of the proposed model, the *accuracy* and the precision[1] and recall[2] of the testing results are measured and presented. Knowing the number of articles in topic i (let it be n_i), we first allowed the system to retrieve n_i topmost similar articles and

Table 1. Image-based and text-based similarity for Corpus One.

		K01		K13		K18		K22	
		Image based	Text based	Image based	Text based	Image based	Text based	Image based	Text based
Group 1	K01	1.0000	1.0000	0.0298	0.0543	0.0537	0.0217	0.0592	0.0589
	K02	0.5395	0.4299	0.0175	0.0477	0.0338	0.0131	0.0286	0.0507
	K03	0.5902	0.5554	0.0475	0.0318	0.0496	0.0227	0.0691	0.0649
	K04	0.4501	0.4410	0.0645	0.0281	0.0254	0.0135	0.0377	0.0636
	K05	0.8664	0.8459	0.0189	0.0443	0.0421	0.0209	0.0551	0.0505
	K06	0.3987	0.4112	0.0398	0.0681	0.0235	0.0087	0.0775	0.0633
	K07	0.3877	0.3993	0.0071	0.0297	0.0295	0.0264	0.0194	0.0301
	K08	0.4448	0.4303	0.0386	0.0313	0.0532	0.0194	0.0964	0.0723
	K09	0.6618	0.6296	0.0118	0.0397	0.0434	0.0395	0.0643	0.0524
	K10	0.0894	0.2463	0.1761	0.1290	0.0551	0.0652	0.0480	0.0572
	K11	0.3607	0.4186	0.0533	0.0348	0.0420	0.0225	0.0208	0.0424
	K12	0.7460	0.6635	0.0080	0.0532	0.0256	0.0191	0.0145	0.0414
Group 2	K13	0.0298	0.0543	1.0000	1.0000	0.0452	0.0474	0.0248	0.0191
	K14	0.0484	0.0348	0.3918	0.3491	0.0413	0.0299	0.0272	0.0336
	K15	0.0472	0.0333	0.3299	0.2959	0.0374	0.0454	0.0431	0.0580
	K16	0.0260	0.0209	0.4219	0.3644	0.0390	0.0773	0.0119	0.0313
	K17	0.0368	0.0347	0.3027	0.3559	0.0182	0.0441	0.0179	0.0141
Group 3	K18	0.0537	0.0217	0.0452	0.0474	1.0000	1.0000	0.0547	0.0594
	K19	0.0740	0.0562	0.0457	0.0263	0.1696	0.2917	0.0624	0.0437
	K20	0.0418	0.0326	0.2217	0.0344	0.1089	0.2055	0.0176	0.0380
	K21	0.0293	0.0257	0.0262	0.0351	0.0767	0.2102	0.0373	0.0526
Group 4	K22	0.0592	0.0589	0.0248	0.0191	0.0547	0.0594	1.0000	1.0000
	K23	0.0334	0.0573	0.0448	0.0217	0.0582	0.0607	0.2372	0.2808
	K24	0.0420	0.0448	0.0406	0.0177	0.0342	0.0412	0.3994	0.4700
	K25	0.1460	0.1073	0.0852	0.0299	0.0381	0.0370	0.1255	0.1834
	K26	0.0204	0.0648	0.0566	0.0139	0.0270	0.0323	0.1278	0.1761

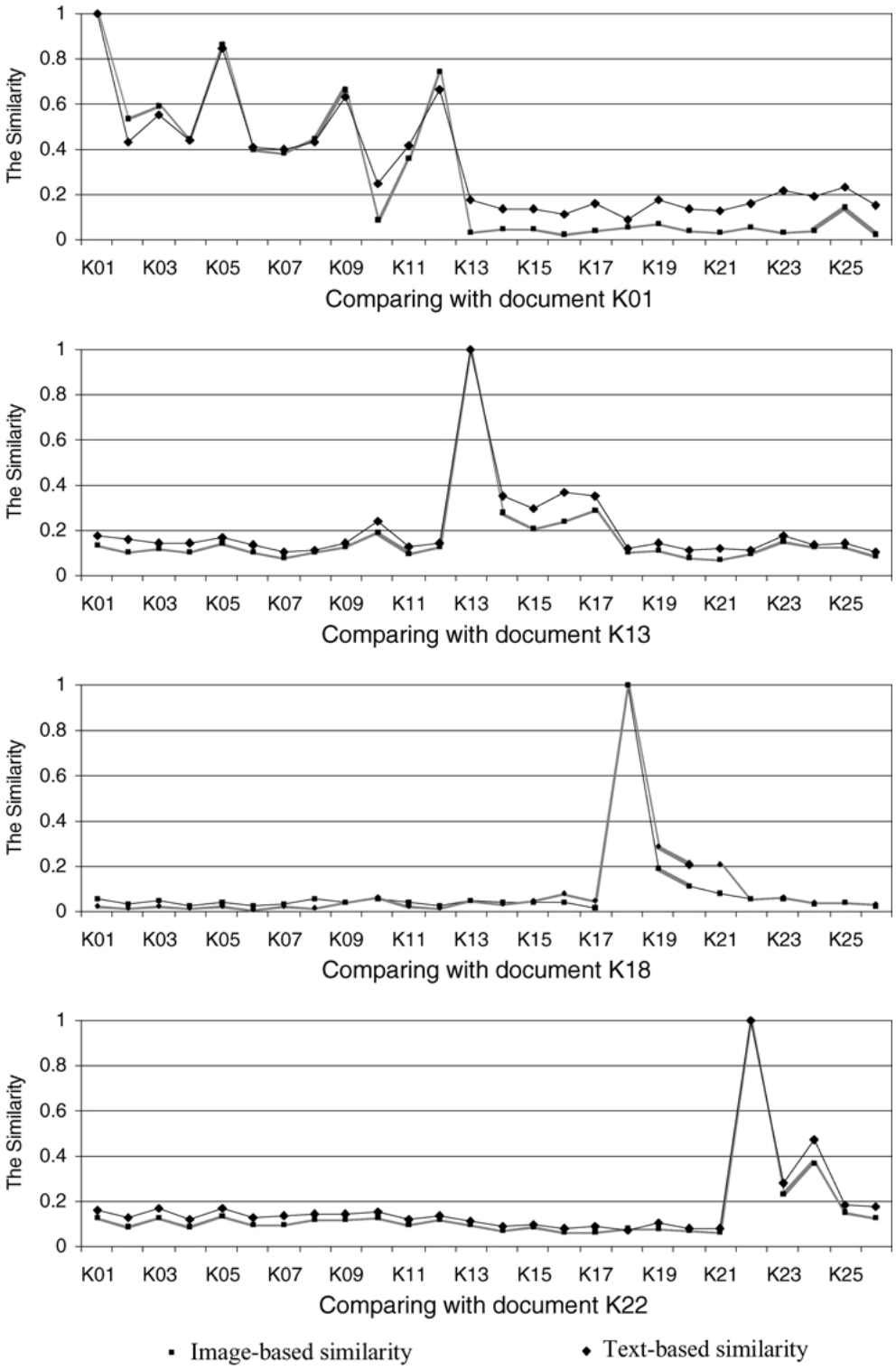
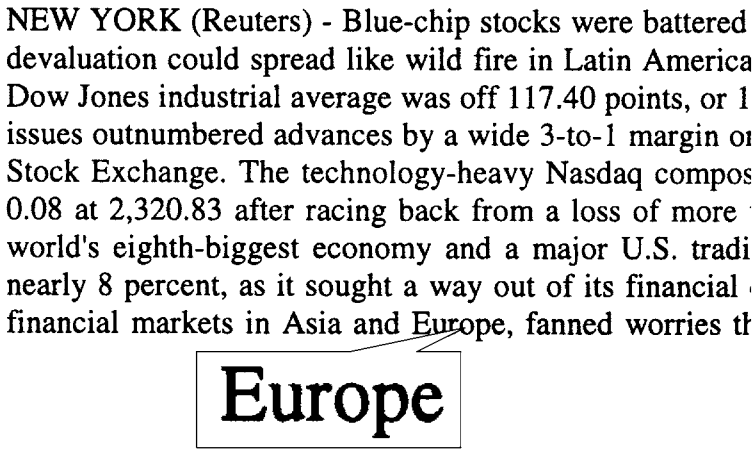
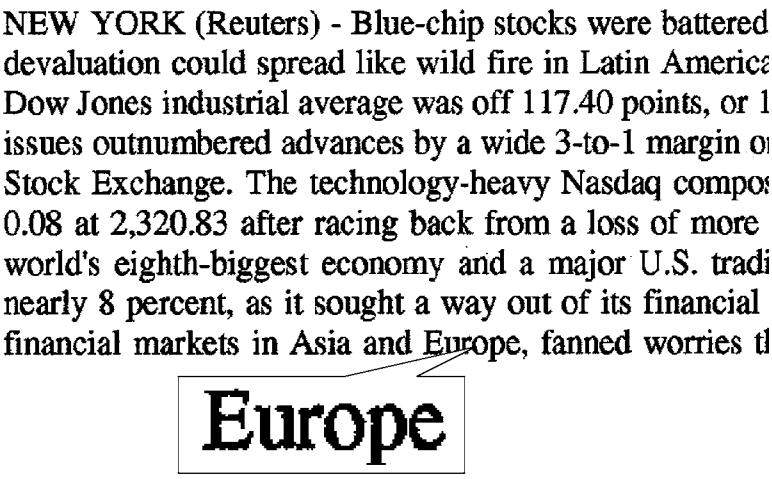


Figure 6. Comparison of image-based and text-based similarity for Corpus One.



(a) Sample document with resolution 600 pixels/inch (ppi)



(b) Sample document with resolution 300 pixels/inch (ppi)

Figure 7. Sample articles in different resolutions from Corpus Two. The enlarged words show the detailed differences in quality between the two documents.

determined how many of these n_i articles are about topic i. Let this number of correctly retrieved articles be m_i. We define *accuracy* of this retrieval process as m_i/n_i. We next retrieved articles based on the threshold instead of a pre-determined number of articles. We set threshold at 0.1, 0.15 and 0.2 in the following experiments respectively, and find the values of precision and recall.

To examine the effectiveness of the proposed method handling degradation of documents, Corpus Two consisting of 52 document images was generated using the same set of documents as in Corpus One. Document images T01–T26 were obtained by scanning at 600 pixels/inch (ppi) and documents images R01–R26 were scanned in with resolution of 300 pixels/inch (ppi) to simulate the degradation of image quality. Two sample articles in the two resolutions are shown in Fig. 7(a) and (b) respectively. In Fig. 7(a), characters in the article have clear edges and are separated from each other, while the edges of characters in Fig. 7(b) are noisy and characters touch each other more often. The enlarged word "Europe" in both images shows the effect of degradation. In this way, we can observe the performance of the proposed method under different image qualities. Within each resolution group, we performed similarity measure in a similar way as in Corpus One. We extracted documents according to their similarity values, and then obtained the accuracy, recall

Table 2. Performance evaluation for degraded documents in Corpus Two.

			Threshold = 0.2		Threshold = 0.15		Threshold = 0.1	
Document type	Query article	Accuracy %	Recall %	Precision %	Recall %	Precision %	Recall %	Precision %
Image based resolution 600	T01	91.67	91.67	91.67	100	80	100	52.17
	T13	100	100	83.33	100	50	100	29.41
	T18	75	25	100	25	100	75	50
	T22	60	40	100	40	100	100	62.5
	Average	**81.67**	**64.17**	**93.75**	**66.25**	**82.5**	**93.75**	**48.52**
Image based resolution 300	R01	91.67	83.33	90.91	100	92.31	100	63.16
	R13	40	40	66.67	60	37.5	100	31.25
	R18	50	25	100	25	100	50	33.33
	R22	80	40	100	60	100	80	66.67
	Average	**65.42**	**47.08**	**89.40**	**61.25**	**82.45**	**82.5**	**48.60**
Text based	O01	91.67	91.67	100	91.67	91.67	100	75
	O13	100	100	100	100	100	100	83.33
	O18	75	50	100	75	100	100	66.67
	O22	80	60	100	60	100	60	75
	Average	**86.67**	**75.42**	**100**	**81.67**	**97.92**	**90**	**75**

and precision. The comparisons of the accuracies, recalls and precisions between two different resolutions are summarized in Table 2. As a benchmark, the result of text-version N-gram based method is also shown in Table 2.

Corpus Three was generated to test the robustness of the proposed method dealing with documents in different fonts. We used the same set of 26 articles as in Corpus One to generate document images using four different fonts, namely the Arial (A01–A26), Lucida Sans (L01–L26), Myriad Roman (M01–M26) and Times New Roman (T01–T26). Thus 104 documents with different fonts constitute our corpus Three. Sample articles in these four fonts are shown in Fig. 8(a)

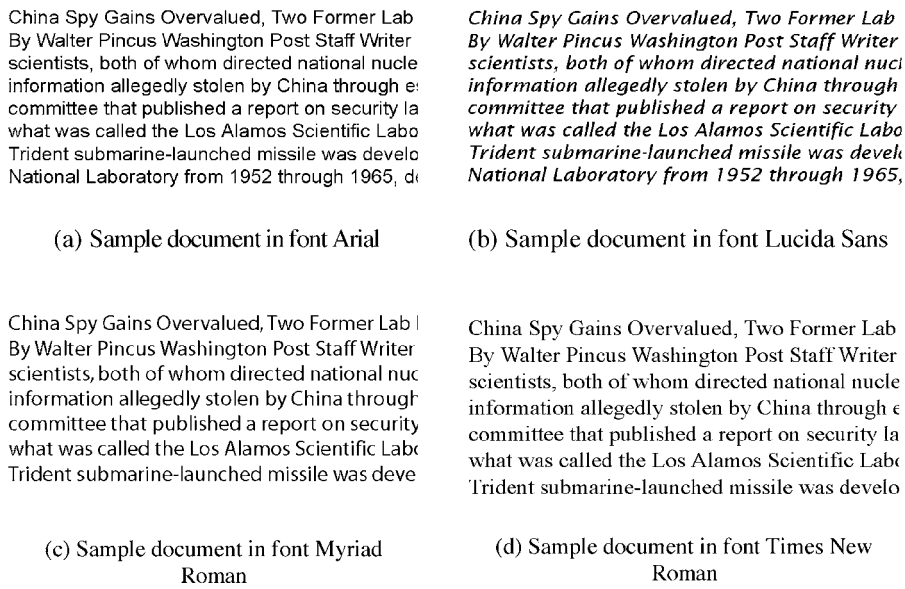


Figure 8. Sample documents in four different fonts from Corpus Three.

Table 3. Performance evaluation for documents in mixed fonts in Corpus Three.

			Threshold = 0.2		Threshold = 0.15		Threshold = 0.1	
Document type	Query article	Accuracy %	Recall %	Precision %	Recall %	Precision %	Recall %	Precision %
Image based	A01	85.42	52.08	96.15	66.67	94.12	87.5	66.67
	A13	65	45	84.82	50	76.92	65	38.24
	A18	31.25	6.25	100	6.25	100	31.25	33.33
	A22	55	45	75	55	44	85	33.33
	L01	64.58	54.17	83.87	64.58	72.09	85.42	53.25
	L13	60	70	56	75	38.46	100	28.17
	L18	31.25	6.25	100	12.5	50	31.25	29.41
	L22	65	40	80	70	53.85	80	34.78
	M01	81.25	35.42	100	54.17	92.86	81.25	73.58
	M13	70	30	100	50	83.33	70	63.64
	M18	31.25	12.5	100	12.5	100	25	66.67
	M22	45	35	58.33	55	50	75	27.78
	T01	83.33	62.5	96.77	83.33	83.33	95.83	64.79
	T13	65	55	61.11	80	44.44	95	31.15
	T18	31.25	12.5	100	25	57.14	31.25	20.83
	T22	50	30	60	50	50	70	28.57
	Average	**57.16**	**36.98**	**84.32**	**50.63**	**68.16**	**69.30**	**43.39**
Text based	O01	91.67	91.67	100	91.67	91.67	100	75
	O13	100	100	100	100	100	100	83.33
	O18	75	50	100	75	100	100	66.67
	O22	80	60	100	60	100	60	75
	Average	**86.67**	**75.42**	**100**	**81.67**	**97.92**	**90**	**75**

to (d). We measured similarities among all documents in the corpus and used the first document from each topic in each font as the query article to obtain the accuracies, recalls and precisions, which are shown in Table 3. Again the results for the text-based N-gram method are also shown in Table 3 for comparison.

From the experiment results obtained for Corpus One, we can see that the similarities of documents measured from text-mode articles using traditional N-gram method and image-based articles using the proposed method share some resemblance though not entirely equivalent to each other. The result of the text version of documents provides more distinguishable similarity measures. This is because the vertical bar patterns extracted from the document images are not as distinguishable as words themselves in text documents due to the collisions. The collision here is defined as the same vertical bar pattern resulted from different words. The number of collisions is inversely proportional to the length of the vertical bar pattern, thus we can reduce the effect of the collisions by applying a lower bound to the length of vertical bar pattern to be compared. The image-based similarity provides an adequate means to retrieve similar news articles with respect to a query article.

Furthermore, the results from Corpus Two show the robustness of the proposed method under different document qualities. For documents in resolution 600 pixels/inch (ppi), the average accuracy of the proposed method reaches 81.67%. For threshold of 0.2 the average recall and precision are 64.17% and 93.75% respectively, while for threshold of 0.1 the average recall and precision are 93.75% and 48.52% respectively, reflecting the trade-off between the recall and precision. When the resolution of the document drops by half, the average accuracy drops by 15.25% and the recall and precision decrease to a certain extent depending on the choice of threshold. As the threshold value becomes smaller, the decrease in recall and precision is less obvious. Actually for threshold value of 0.15, the recall and precision values for the documents in two resolutions are very close to each other. This suggests that

the performance of the proposed method is not affected much for degraded documents.

From the sample images shown in Fig. 8, the changing of document fonts results in different character shapes, different character size and different cases of character touching. However the results from Corpus Three still show the encouraging ability of the proposed method to handle mixture of documents in different fonts. The average accuracy reaches 57.16%. For threshold of 0.2 the average recall and precision are 36.98% and 84.32% respectively, while for threshold of 0.1 the average recall and precision become 69.30% and 43.39% respectively. We notice that the results for some topics are very good while the results for some other topics are somehow less favorable due to noise and collisions. Overall speaking, the proposed method can provide favorable recall and precision for retrieving documents in different fonts depending on the user's requirement.

To further examine the ability of the proposed method to extract relevant documents, we created Corpus Four containing 159 English news articles that were grouped into eight major topics based on their contents. These articles, being different from the first three corpora, are images either downloaded from the web or scanned from newspaper cuttings. The text version documents were obtained through OCR and were corrected by hand. The first article in each of the eight topics was selected as the query document and was compared with all documents in the corpus. The resulting accuracies, recalls and precisions are recorded and compared with the corresponding results obtained by text-version N-gram based method in Table 4. Besides the precision and recall, we also use the composite measure F_1 [31] defined below as sometimes used in the information retrieval community:

$$F_1 = \frac{2RP}{R+P}$$

The F_1 measure is derived from the recall and precision. It is a strict measurement because it does not only reflect the absolute value of the recall and precision but also reflects the degree of balance between the two. We calculated the F_1 values for both the image-based method and the text-based method and the results are presented in Table 4.

It can be seen that the average accuracy for Corpus Four is 57.76%. At the threshold of 0.2, the average recall and precision are 48.94% and 67.79%, respectively, whereas choosing 0.1 as the threshold will give

Table 4. Accuracy, recall, precision and F_1 of image-based document text retrieval for Corpus Four.

Document type	Topic id. no. i	NR	Accuracy %	Threshold = 0.2			Threshold = 0.15			Threshold = 0.10		
				P %	R %	F_1 %	P %	R %	F_1 %	P %	R %	F_1 %
Image based	1	26	57.69	100	34.62	51.43	68.18	57.69	62.50	42.31	84.62	56.41
	2	22	27.27	33.33	22.73	27.03	25.81	36.36	30.19	25.86	68.18	37.50
	3	18	61.11	60	66.67	63.16	45.45	83.33	58.82	23.94	94.44	38.20
	4	16	68.75	76.92	62.5	68.96	68.75	68.75	68.75	57.14	75	64.86
	5	22	13.64	28.57	9.09	13.79	12	13.64	12.77	14.47	50	22.44
	6	19	94.74	100	73.68	84.85	100	78.95	88.24	100	94.74	97.30
	7	18	50	43.48	55.56	48.78	35.45	61.11	44.89	19.70	72.22	30.96
	8	18	88.89	100	66.67	80.00	100	83.33	90.91	100	88.89	94.12
	Average		**57.76**	**67.79**	**48.94**	**54.75**	**56.96**	**60.40**	**57.13**	**47.93**	**78.51**	**55.22**
Text based	1	26	88.46	100	50	66.67	87.5	80.77	84.00	70.59	92.31	80.00
	2	22	95.46	100	86.36	92.68	100	95.45	97.67	41.51	100	58.67
	3	18	100	99	100	94.74	41.86	100	59.02	16.07	100	27.69
	4	16	100	88.89	100	94.11	48.48	100	65.30	14.29	100	25.01
	5	22	27.27	33.33	4.55	8.01	17.65	13.64	15.39	25.61	95.45	40.38
	6	19	100	100	73.68	84.85	100	100	100	100	100	100
	7	18	94.44	100	72.22	83.87	93.75	83.33	88.23	53.12	94.44	67.99
	8	18	100	100	100	100	100	100	100	27.27	100	42.85
	Average		**88.20**	**89.03**	**73.35**	**78.12**	**73.66**	**84.15**	**76.20**	**43.56**	**97.78**	**55.32**

NR = Number of relevant documents in the group, P = Precision, R = Recall.

Table 5. Comparison of total processing time between the proposed method and the OCR approach.

Method	Corpus One (26 docs)	Corpus Two (52 docs)	Corpus Three (104 docs)	Corpus Four (159 docs)
OCR	6 min	12 min	24 min	36 min
Vertical Bar	1 min	2 min	10 min	11 min

an average recall and precision of 78.51% and 47.93%, respectively. Thus, setting a higher threshold gives a better precision but poorer recall, and the reverse is true for a lower threshold. If the emphasis is on retrieving only relevant articles, then a 0.2 threshold should be used. On the other hand, if the intent is to retrieve as many as possible news articles, then a threshold of 0.1 may be adopted. We can see that the average F_1 value is 54.75% for threshold of 0.2 and 55.22% for threshold of 0.1. On the other hand, the F_1 value for threshold of 0.15 that is 57.13% which is higher than both for threshold 0.2 and 0.1. This indicates that threshold 0.15 may be a good compromise. Again, we observed outstanding performance of the proposed method for some topics such as topics 6 and 8. But we also observed relatively poor results for some other topics such as topics 2 and 5. Surprisingly we found that the result obtained for topic 5 using the text-based N-gram method is also poor, appearing as an exception for the statistical similarity measure.

Finally we took some measurement of the processing time in comparison with an OCR process. During the experiments using the four corpora, we recorded down the processing time for the proposed method. As a benchmark, we used an OCR system to convert documents to texts and then performed N-gram based similarity measure. The total processing time of the two approaches are summarized in Table 5. From the result we can see that for all the four corpora the OCR system takes longer time to generate the similarity values among document images. This is because the OCR approach has an extra step to convert images into texts. Of course an advantage for the OCR approach is that the text version of the documents can be obtained. However when speed is an important concern, the proposed method is more suitable than the OCR approach.

6. Conclusion and Future Work

A new model of document image text retrieval based on an image-based similarity measurement without the use of OCR is proposed in this paper. Features called the vertical bar patterns are extracted from word units in a document image through local extrema points detection, and form the document vector. Document similarity is calculated by finding the scalar product between two vectors. Experiments using four corpora of news articles have confirmed the validity of the model for extracting relevant articles among groups of documents, including degraded documents and documents in mixed fonts. Due to its faster processing, the present method offers a speedier way of retrieving scanned images based on content similarity instead of going through an OCR process. If only a handful of relevant imaged documents are needed out of a huge image database, to do OCR on the entire database before retrieving will be a wasteful effort.

The method is suitable for gauging the similarity of document images written in Latin languages. An issue that may be examined in the future is how the removal of special accent characters in some European languages affects the similarity measure. Another research direction is to extend the current method to handle documents in other languages. The final objective of our method is to use it in the retrieval of document images in National University of Singapore's Digital Library. A century old collection of Chinese newspapers currently forms a major part of the archived document base. Text similarity based on Chinese character shape analysis is the next task in our agenda. Similarly, Tamil (a dominant Indian language used in Singapore) character shape analysis can also be studied.

We can convert ASCII text documents into vertical bar patterns by a proper lookup table. In this way, the present text retrieval method based on text similarity can be applied to a mixture of imaged documents and ASCII text documents, as well as to a range of different languages.

Acknowledgments

This project is supported by the research grant R252-000-071-112/303 from the National Science and Technology Board and Ministry of Education of Singapore.

Notes

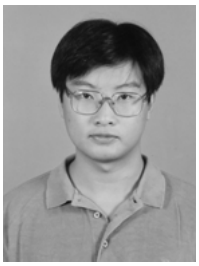
1. Precision is defined as percentage of the number of correctly retrieved articles over the number of all retrieved articles.
2. Recall is defined as percentage of the number of correctly retrieved articles over the number of articles in the category.

References

1. G. Salton, "Developments in automatic text retrieval," *Science*, vol. 253, pp. 974–980, 1991.
2. G. Salton and C. Buckley, "Global text matching for information retrieval," *Science*, vol. 253, pp. 1012–1015, 1991.
3. G. Salton, J. Allan, C. Buckley, and A. Singhal, "Automatic analysis, theme generation, and summarization of machine-readable text," *Science*, vol. 264, pp. 1421–1426, 1994.
4. C.E. Shannon, *The Mathematical Theory of Communication*, University of Illinois Press: Urbana, 1949.
5. C.Y. Suen, "N-gram statistics for natural language understanding and text processing," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, *PAMI*, vol. 1, no. 2, pp. 164–172, 1979.
6. A. Zamora, "Automatic detection and correcting of spelling errors in a large data base," *Journal of the American Society for Information Science*, vol. 31, no. 51, 1980.
7. J.L. Peterson, "Computer programs for detecting and correcting spelling errors," *Comm.* vol. ACM 23, p. 676, 1980.
8. E.M. Zamora, J.J. Pollock, and A. Zamora, "The use of trigram analysis for spelling error detection," *Inf. Proc. Mgt.* vol. 17, p. 305, 1981.
9. J.J. Hull and S.N. Srihari, "Experiments in text recognition with binary N-gram and Viterbi algorithms," *IEEE Trans. Pattern Analysis & Machine Intelligence*, vol. PAMI-4, p. 520, 1980.
10. J.J. Pollock, "Spelling error detection and correction by computer: Some notes and a bibliography," *J. Doc.* vol. 38, p. 282, 1982.
11. R.C. Angell, G.E. Freund, and P. Willette, "Automatic spelling correction using trigram similarity measure," *Inf. Proc. Mgt.* vol. 18, p. 255, 1983.
12. E.J. Yannakoudakis, P. Goyal, and J.A. Huggill, "The generation and use of text fragments for data compression," *Inf. Proc. Mgt.* vol. 18, p. 15, 1982.
13. J.C. Schmitt, "Trigram-based method of language identification," U.S. Patent No. 5,062,143, 1990.
14. W.B. Cavnar and J.M. Trenkle, "N-gram-based text categorization," in *Proceeding of the Symposium on Document Analysis and Information Retrieval*, University of Nevada, Las Vegas, 1994.
15. P. Willett, "Document retrieval experiments using indexing vocabularies of varying size. II. Hashing, Truncation. Digram and trigram encoding of index terms," *J. Doc.* vol. 35, p. 296, 1979.
16. W.B. Cavnar, "N-gram-based text filtering for TREC-2," *The Second Text Retrieval Conference (TREC-2)*, NIST Special Publication 500-215, National Institute of Standards and Technology, Gaitherburg, Maryland, 1994.
17. Marc Damashek, "Gauging similarity via N-grams: Language-independent sorting, categorization, and retrieval of text," *Science*, vol. 267, pp. 843–848, 1995.
18. S.M. Harding, W.B. Croft, and C. Weir, "Probabilistic retrieval of OCR degraded text using N-grams," in *European Conference on Digital Libraries*, pp. 345–359, 1997.
19. W.B. Croft, S.M. Harding, K. Taghva, and J. Borsack, "An evaluation of information retrieval accuracy with simulated OCR output," in *Symposium of Document Analysis and Information Retrieval*, pp. 115–126, 1994.
20. F.R. Chen and D.S. Bloomberg, "Extraction of thematically relevant text from images," in *Proceedings of the Symposium on Document Analysis and Information Retrieval*, pp. 163–178, 1996.
21. F.R. Chen and D.S. Bloomberg, "Extraction of indicative summary sentences from imaged documents," in *Proceedings of the Fourth International Conference on Document Analysis and Recognition (ICDAR'97)*, vol. 1, pp. 227–232, 1997.
22. J.J. Hull and J.F. Cullen, "Document image similarity and equivalence detection," in *Proceedings of the Fourth International Conference on Document Analysis and Recognition (ICDAR'97)*, vol. 1, pp. 308–312, 1997.
23. Y. He, Z. Jiang, B. Liu, and H. Zhao, "Content-based indexing and retrieval method of chinese document images," in *Proceedings of the Fifth International Conference on Document Analysis and Recognition (ICDAR'99)*, pp. 685–688, 1999.
24. P. Sibun and A.L. Splitz, "Language determination: National language processing from scanned document images," in *Proceedings of the fourth Conference on Applied Natural Language Processing*, pp. 423–433, Las Vegas, April 1995.
25. A.L. Spitz, "Determination of the script and language content of document images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 235–245, 1997.
26. C.Y. Suen, S. Bergler, N. Nobile, B. Waked, C.P. Nadal, and A. Bloch, "Categorizing document images into script and language classes," in *Proceedings of the International Conference on Advances in Pattern Recognition*, Plymouth, UK, pp. 297–306, 23–25 Nov 1998.
27. C.L. Tan, P.Y. Leong, and S. He, "Language identification in multilingual documents," in *International Symposium on Intelligent Multimedia and Distance Education*, 1999.
28. A.F. Smeaton and A.L. Spitz, "Using character shape coding for information retrieval," in *Proceeding of the Fourth International Conference on Document Analysis and Recognition (ICDAR97)*, vol. 2, pp. 974–978, 1997.
29. R.K. Powalka, N. Sherkat, and R.J. Whitrow, "Word shape analysis for a hybrid recognition system," *Pattern Recognition*, vol. 30, no. 3, pp. 421–445, 1997.
30. D.S. Bloomberg, G.E. Kopec, and L. Dasari, "Measuring document image skew and orientation," *SPIE Conf.* 2422, Document Recognition II, San Jose, CA, pp. 302–316, 1995.
31. Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 42–49, 1999.

Chew Lim Tan is an Associate Professor in the Department of Computer Science, School of Computing, National University of Singapore. His research interests are expert systems, document image and text processing, neural networks and genetic programming. He obtained a B.Sc. (Hons) degree in Physics in 1971 from the

University of Singapore, an M.Sc. degree in Radiation Studies in 1973 from the University of Surrey, U.K., and a Ph.D. degree in Computer Science in 1986 from the University of Virginia, U.S.A.



Weihua Huang is a research scholar in the Department of Computer Science, School of Computing, National University of Singapore. His research interests are word shape recognition and image-based document retrieval. He obtained a B.Sc. (Hons) degree in Computer Science from the National University of Singapore in 2000, and is currently pursuing an M.Sc. degree in the School of Computing, National University of Singapore.

Sam Yuan Sung is an Associate Professor in the Department of Computer Science, School of Computing, National University of Singapore. His research interests include information retrievals, data mining, pictorial databases and mobile computing. He received a B.Sc. degree from National Taiwan University in 1973, an M.Sc. degree and a Ph.D. degree in Computer Science from the University of Minnesota in 1979 and 1982, respectively. He was with the University of Oklahoma and Memphis State University in USA before joining the National University of Singapore.

Zhaohui Yu obtained a B.S. degree in Computer Software from the National University of Defence Technology, Changsha in China in 1991 and an M.Sc. degree in Computer Science from the National University of Singapore in 2001. His main research interest is in document image processing. He is currently a software engineer in ATI Technologies Inc., Canada.

Yi Xu was a Research Assistant in the Department of Computer Science, School of Computing, National University of Singapore. He obtained a B.E. degree from the University of Science and Technology of China in 1984. He worked in the Remote Sensing Satellite Ground Station, Chinese Academy of Sciences from 1989 to 1998 before joining the National University of Singapore. He now works in Agilent Technologies Inc., Singapore.